NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

DISSERTATION

AN ADAPTIVE, MULTI-RATE LINEAR QUADRATIC REGULATOR FOR A SHIPBOARD MVDC DISTRIBUTION SYSTEM WITH CONSTANT POWER LOADS

by

Adam J. Mills

September 2017

Dissertation Supervisor: Robert W. Ashton

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		*Form Approved OMB No. 0704–0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE September 2017	3. REPORT TYPE AND DATES COVERED Dissertation	
4. TITLE AND SUBTITLE AN ADAPTIVE, MULTI-RATE LINEAR QUADRATIC REGULATOR FOR A SHIPBOARD MVDC DISTRIBUTION SYSTEM WITH CONSTANT POWER LOADS		5. FUNDING NUMBERS **REL95** **REK4K**	
6. AUTHOR(S) Adam J. Mills			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB number ____N/A____.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.		12b. DISTRIBUTION CODE	

13. ABSTRACT (maximum 200 words)

Future U.S. Navy warships will have DC electrical distribution systems. These distribution systems will include multiple layers of electronic power converters. When coupled to high-bandwidth controllers, power converters behave as constant power loads to the distribution system. Constant power loads have a negative non-linear impedance characteristic that threatens system stability.

Many different single-input control schemes have been applied to DC microgrids with constant power loads. In this work we propose a centralized select-matrix, multi-rate linear quadratic regulator (LQR-SM)-based control scheme to provide a flexible and adaptable controller for high-order, multi-input, and multi-rate distribution systems. The proposed controller is investigated via MATLAB time-domain simulation. LQR-SM controller performance is compared to both block-cyclic multi-rate LQR and state-feedback linearization. LQR-SM controller simulations show vastly reduced computational load and improved robustness compared to block-cyclic LQR and reduced energy-storage element sizing compared to state-feedback linearization.

A genetic algorithm design procedure is described for the controller. The design process outlines evaluation function development, choosing the initial generation of candidates, and genetic algorithm process flow.

Finally, engineering trade-offs are considered. In this work we investigate engineering trade-offs with respect to computational load, regions of attraction, and energy-storage efficiency when reduced fidelity and non-adaptive controller configurations are considered.

14. SUBJECT TERMS medium voltage DC, constant power loads, CPL, linear quadratic regulator, LQR, integrated power systems, IPS, multirate, genetic algorithm			15. NUMBER OF PAGES 145
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU

THIS PAGE INTENTIONALLY LEFT BLANK

AN ADAPTIVE, MULTI-RATE LINEAR QUADRATIC REGULATOR FOR A SHIPBOARD MVDC DISTRIBUTION SYSTEM WITH CONSTANT POWER LOADS

Adam J. Mills
Lieutenant Commander, United States Navy
B.S., University of Washington, 2004
M.S., University of Washington, 2005

Submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

September 2017

Approved by: Robert W. Ashton   Roberto Cristi
              Associate Professor of   Professor of Electrical
              Electrical and Computer   and Computer Engineering
              Engineering
              Dissertation Supervisor

              Xiaoping Yun   Keith Cohn
              Professor of Electrical   Research Professor of Physics
              and Computer Engineering

              Alexander Julian, PhD
              Power Engineering Consultant

Approved by: R. Clark Robertson,
              Chair, Department of Electrical and Computer Engineering

Approved by: Douglas Moses,
              Vice Provost of Academic Affairs

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Future U.S. Navy warships will have DC electrical distribution systems. These distribution systems will include multiple layers of electronic power converters. When coupled to high-bandwidth controllers, power converters behave as constant power loads to their distribution systems. Constant power loads have a negative non-linear impedance characteristic that threatens system stability.

Many different single-input control schemes have been applied to DC microgrids with constant power loads. In this work, we propose a centralized select-matrix, multi-rate linear quadratic regulator (LQR-SM)-based control scheme to provide a flexible and adaptable controller for high-order, multi-input, and multi-rate distribution systems. The proposed controller is investigated via MATLAB time-domain simulation. LQR-SM controller performance is compared to both block-cyclic multi-rate LQR and state-feedback linearization. LQR-SM controller simulations show vastly reduced computational load and improved robustness compared to block-cyclic LQR and reduced energy-storage element sizing compared to state-feedback linearization.

A genetic algorithm design procedure is described for the controller. The design process outlines evaluation function development, choosing the initial generation of candidates, and genetic algorithm process flow.

Finally, engineering trade-offs are considered. In this work, we investigate engineering trade-offs with respect to computational load, regions of attraction, and energy-storage efficiency when reduced fidelity and non-adaptive controller configurations are considered.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AC	alternating current
ATG	auxiliary turbine generator
CCM	continuous conduction mode
DC	direct current
CPL	constant power load
EKF	extended Kalman filter
EMRG	electromagnetic rail gun
HESS	hybrid energy storage system
IEEE	Institute of Electrical and Electronics Engineers
IPS	integrated power system
LQG	linear quadratic Gaussian
LQR	linear quadratic regulator
LQR-PD	periodic-discrete LQR
LQR-SM	select-matric LQR
LVDC	low-voltage DC
LSF	state-feedback linearization
LTI	linear time-invariant
MTG	main turbine generator
MVDC	medium voltage DC
NAVSEA	U.S. Naval Sea Systems Command
PGM	power generation module
RC	resistor-capacitor
RL	resistor-inductor
ROA	region of attraction
UPS	uninterruptible power supply

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Integrated propulsion systems are the future of naval surface combatants. For generations, U.S. Navy ships have been designed and built around the concept of separate power generation and distribution systems for electric power and propulsion power. Demand for electrical power on ships has been growing, fueled by the proliferation of sophisticated sensors, communications equipment, and computing power. With the additional development of energy weapons systems such as lasers and naval railgun, demand for electrical power on combatant craft will rival or exceed demand for propulsion power. If the traditional paradigm of separate electrical and propulsion systems is maintained, the physical size, weight and cost of shipboard engine rooms will become prohibitive [1].

Integrated power systems (IPS) use a common electrical generation and distribution system to supply all loading on the ship. No matter whether the loads are typical hotel fare, like lighting and air conditioning, energy weapons, or propulsion drives, all loads are powered from a common distribution system. An IPS takes advantage of typical power profiles for propulsion, ship's service, and combat system loads to appropriately size the capacity of the generators. Typically, full propulsion loading is only used when the ship is transiting between patrol areas, and low speeds are used during patrol. Combat systems are not typically used during transit periods. Since not all loads will be on-line simultaneously, these historical operating norms enable electricity generating capacity to be smaller than total load capacity while still meeting the needs of the ship [2]. IPS architectures have been installed on commercial cruise ships and other vessels, but the first instance of installation of an IPS on a surface combatant is the U.S. Navy destroyer USS *Zumwalt* (DDG-1000). The USS *Zumwalt*'s IPS is a 4160 V 60 Hz AC distribution system. While the capabilities of this platform are extraordinary, the next generation of IPS warship is expected to have a medium voltage DC distribution system. USS *Zumwalt* has a large number of transformers and power converters in order to distribute power to meet various different load needs. So far, *Zumwalt*'s 4160 V 60 Hz AC power must be converted to 450 V 60 Hz AC, 120 V 60 Hz

AC, 120 V 400 Hz AC, and low voltage DC power. The motivation for switching to a DC IPS architecture is that size, cost, and weight savings will be realized by eliminating transformers, reducing the size and weight of cable runs, and minimizing the number and type of power conversions. Furthermore, DC distribution eliminates the need to match frequency and phase when adding another power source to the distribution bus. The DC distribution also improves acoustic performance since generators and motors are no longer synchronized to 60 Hz harmonics. Additionally, the U.S. Navy expects to find performance gains by consolidating and centralizing energy storage. Currently, critical devices are protected from power outages by uninterruptible power supplies (UPS). Each UPS is sized to meet worst-case conditions. By moving to a DC IPS, energy storage can be moved from point-of-use to points-of-distribution within protected zones. The consolidation of UPS into fewer and larger energy storage devices should result in lower cost, less space consumed, and more efficient power conversion [2].

A. MOTIVATION

A DC distribution system will inevitably have many electronics-based power converters. Electronic power converters are typically coupled with control systems to regulate output current and voltage or input impedance; thus, power converters do not exhibit linear impedance to the distribution system in the same way that passive loads do. In the worst-case scenario, tightly controlled power converters behave as constant power loads (CPL). That is to say that regardless of input voltage to the power converter, the power consumed by the load is constant. CPL have negative non-linear impedance. When the net impedance of the distribution bus is zero or negative, the distribution system becomes unstable [3]. Since the future MVDC warship will very likely have most, if not all, of its loads interfaced and controlled through tightly regulated power converters, assuring stability of the distribution system is a primary concern.

Incorporating energy storage into the distribution system is also a new challenge. Traditionally, UPS and other types of energy storage devices have been used only in "off-line" configurations. The energy storage devices are activated when a loss of primary power is detected. When primary power returns, the energy storage device reverts back to

a charging/standby mode. This paradigm for using energy storage devices is inefficient. Hybrid energy storage systems (HESS) are energy storage devices that seek to combine the advantages of high power-density devices such as capacitors and high energy-density devices such as batteries and flywheels by combining the devices with power electronics and control systems [4]. By keeping the HESS connected to the power bus while main power is connected, the dynamic advantages of the HESS can be used to regulate the electrical distribution system. In one example, a HESS was used in a hypothetical shipboard power system to cancel harmonics introduced by wave motion on the propulsion motor [5]. In another example, HESS are a part of energy management schemes to allow total system loading to exceed generating capacity. In these cases, whenever total electrical loading exceeds generating capacity, the HESS provide power to fill the deficit [6], [7]. While the energy management schemes in [6] and [7] are largely interested in using the HESS to provide ride-through power when pulsed loads exceed generating capacity, others have explored using HESS as a means to improve voltage regulation at all times. In essence, HESS become part of the bus regulation control strategy.

B. OBJECTIVE

The primary goal of this research is to develop a control scheme to regulate the bus voltage in a hypothetical naval MVDC electrical distribution system. The hypothetical naval MVDC distribution system will be derived from concepts published by the U.S. Navy's Naval Sea Systems Command (NAVSEA). While the control scheme will be applied to a specific hypothetical example, the scheme itself should be generalizable to a wider class of problems. Furthermore, we wish to present a design strategy for our controller such that no part of the design process is hidden from the reader. Any engineer should be able to read this document, follow the methods presented, and obtain a satisfactory result.

While previous authors have explored manifold methods to improve stability in systems with CPL, we have found their work inadequate for our application. We have discovered no sources within the IEEE Xplore archive which have explored a multi-input

approach to stabilizing multi-machine MVDC ships or microgrids. We wish to coordinate the actions of power generation unit voltages and HESS currents in order to ensure system stability under steady-state conditions as well as during transients caused by large step-changes in CPL loading.

The secondary goal of our control scheme is to minimize the volume, weight, and acquisition cost of the control scheme. As the literature review shows, stability can be assured through use of various means. Many of these means, such as adding large quantities of capacitance or introducing passive resistances, are quite inefficient.

C.   OUTLINE OF DISSERTATION

The outline of this dissertation is as follows: background information, such as the electric warship concept, DC-DC converters, average value models, constant power loads, and literature review of state-of-the-art controllers for this application are discussed in Chapter II. In Chapter III, we use the information covered in Chapter II to construct a hypothetical circuit model for a naval MVDC IPS electrical distribution system. The proposed control method is described in Chapter IV. The design approach for the proposed controller is outlined in Chapter V. In Chapter VI, we explore performance trade-offs incurred from different controller design approaches. Conclusions and future work are presented in Chapter VII.

II. BACKGROUND

A. THE ELECTRIC WARSHIP

The Electric Warship [8] is a concept for a naval surface warship for which the electrical distribution system is the central enabler of combat power. The key technologies that enable the Electric Warship are power electronics, control systems, high power-density electric motors, and energy weapon systems.

Power electronics allow power to be generated at one voltage and frequency, transmitted at another, and then consumed by a load at yet another voltage and frequency. By interspersing power electronics between the generation, transmission, and consumption stages in a distribution system, we can optimize each different stage according to the needs of that particular stage. Advanced electric drive motors and motor drives are key enablers since we use these machines for high power propulsion in our electric ship. Developing and acquiring high power-density machines and reliable drives is absolutely necessary for installation on board a naval asset. Control systems tie the elements of the system together. Specifically, the power electronics require high-speed, high reliability controllers to direct the action of semiconductor switches.

IPS architecture integrates the propulsion system with the electrical distribution system. In conventional naval architectures, the electrical system and propulsion system are independent systems. The propulsion train has prime movers connected to the propulsion shafts via mechanical transmissions. In large ships, such as aircraft carriers, this arrangement has propulsion shafts running the entire 1100-ft length of the ship, penetrating several watertight bulkheads. In an IPS, power available from the distribution system is converted by a motor drive into the form required by the electrical propulsion motors. The propulsion motors can be placed in one compartment, the motor drive can be placed in a different compartment, and the power generating unit can be in another.

The flexibility of an IPS has many advantages. First, prime mover speed, distribution system electrical frequency, and propulsion motor speed are no longer connected, improving acoustic performance. Second, propulsion and hotel loads can

share power generating units. This allows prime movers to be brought on line or secured to better match generating capacity to loading. Matching generating power to loading can be more efficient than the standard arrangement because the gas turbines used for prime movers are more efficient when they are fully loaded. By amalgamating loads onto fewer machines, greater fuel efficiency can be achieved. Second, sharing power generation by propulsion loads, combat system loads, and hotel loads can reduce the number of prime movers while still maintaining redundancy. Reducing the number of prime movers reduces acquisition cost, maintenance, and saves space and weight. A third major advantage is ship arrangement flexibility. By eliminating the mechanical connection between prime movers and propulsion shafting, we can reduce shaft lengths and reallocate compartment space for other purposes. Prime movers can be redistributed to compartments above the waterline for better flooding protection. This allows exhaust stacks to be much shorter and take up less internal volume within the ship, as shown in Figure 1. In Figure 1, we can see that engineering spaces are designated in blue, freely configurable space is shaded in brown, engine exhaust paths are shaded in pink, while propulsion shafting is a red line.
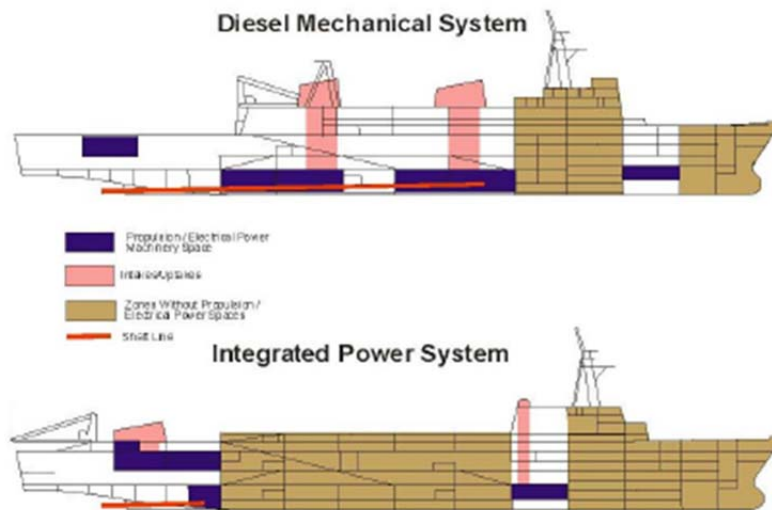
Figure 1. IPS Arrangement Comparison. Source: [9].

B.   DC-DC CONVERTERS

Power converters can convert energy between alternating current and direct current (AC-DC), between one DC voltage level and another (DC-DC), and even between one AC energy type and another (AC-AC). These electronics are largely based on high-speed, high-power rating semiconductor switches. Power electronics are active devices which require control mechanisms to regulate the switching action of the semiconductor devices.

The power converter we discuss here, for illustration purposes, is the buck converter. The buck converter is a DC-DC converter that converts DC power from a higher voltage to a lower voltage. There are also boost converters that convert energy from lower voltage to higher voltage, and buck-boost converters which are able to both buck and boost energy. Academic literature presents a wide variety of topologies, each with its own particular benefits and applications. Rather than delve into specifics, we review a very basic buck converter since doing so is instructive toward the development of our experiment model. The model of a buck converter circuit, also called buck chopper, is shown in Figure 2.
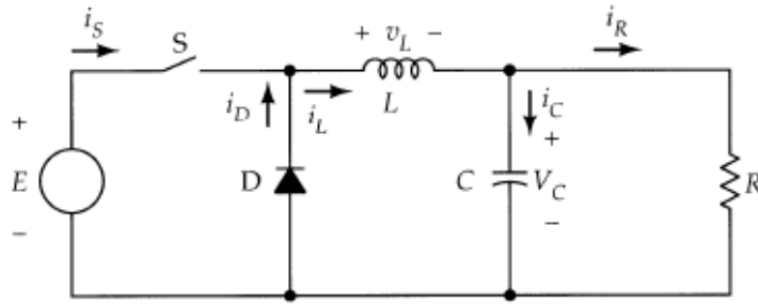
FIGURE 7.1

Figure 2. DC-DC Buck Converter. Source: [10].

When the switch S is shut, the diode D does not conduct. Current flows from the voltage source E through the inductor to the capacitor C and resistive load R. During this period, both the inductor and capacitor can build up stored energy. When the switch S is

7

open, no power flows from the power source, E. Instead, energy for the load comes from the stored energy in both the inductor and capacitor. The two equivalent circuits for switch "shut" and switch "open" are displayed in Figure 3. The differential equations for Figure 3a, the switch "shut" case, are given by

$$\dot{i}_L = \frac{E}{L} - \frac{V_C}{L}$$

$$\dot{V}_C = \frac{i_L}{C} - \frac{V_C}{CR}$$

,(2.1)

and the differential equations for Figure 3b, the switch "open" case, are given by

$$\dot{i}_L = -\frac{V_C}{L}$$

$$\dot{V}_C = \frac{i_L}{C} - \frac{V_C}{CR}$$

,(2.2)

where E is the input source voltage, V_C is the capacitor voltage, i_L is the inductor current, and L, R, and C are the inductor, resistor, and capacitor values, respectively. If the period when the switch is "shut" is T_{on} and the period when the switch is open is T_{off}, the total period T is $T_{on} + T_{off}$. For a fixed period T, a duty cycle D is defined as the proportion of the time the switch is "shut"; therefore, D is T_{on}/T. The steady state analysis of [10] shows that in the continuous conduction mode (CCM), the output voltage V_C of the buck converter is equal to the input voltage multiplied by the duty cycle ($V_C = DE$).
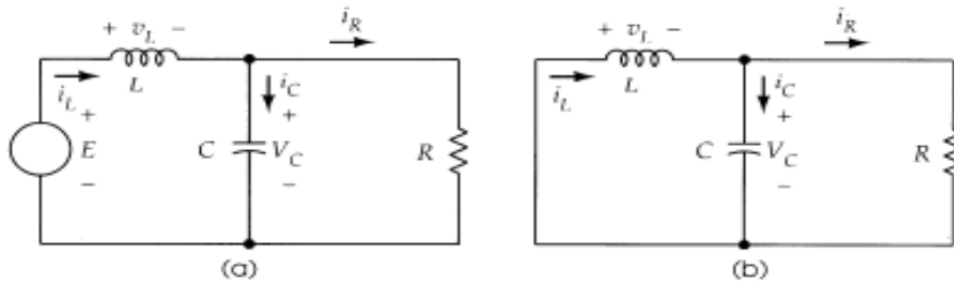
FIGURE 7.2

Figure 3. Buck Equivalent Circuits for Switch "Shut," (a), and "Open," (b)
Source: [10].

8

In order to maintain our buck converter in CCM, the converter must have sufficient inductance for the expected range of load. The minimum inductance the buck converter needs to carry is called the critical inductance. The formula for critical inductance of a buck converter is

$$L_{crit} = \frac{V_C}{2 f_{switch} I_{Load}} (1-D) ,$$

(2.3)

where L_{crit} is the critical inductance, f_{switch} is the frequency of actuating the switch and I_{Load} is the load current. From Equation (2.3), we see that longer duty cycles, higher switching speeds and larger load currents will shrink the size of our inductor; whereas larger loads and lower switching frequencies will do the opposite. We use this critical inductance value later when developing our circuit model.

The next aspect of the buck converter we need to explore is capacitor sizing. Typically, switch-mode power supplies will have a specification for maximum voltage ripple. In order to meet ripple requirements, the filter capacitor C must be sufficiently large. The minimum capacitor size to meet ripple requirements is

$$C_{min} = \left(\frac{1-D}{f_{switch}^2} \right) \left(\frac{100}{8L \cdot \%Vripple} \right) ,$$

(2.4)

where C_{min} is the minimum capacitor size, V_{ripple} is the voltage ripple specification, and all other parameters are as previously defined. In Equation (2.4), we can see that the minimum capacitor size gets smaller for faster switching speed, higher duty cycle, larger inductance, and larger ripple specification.

C.   AVERAGE VALUE MODELS

In order to examine the transient dynamics of the buck converter, we employ the average value model technique. Average value models simply average the differential equations of the switching power converter over one duty cycle. Using this technique, we may "average" the switch out of the circuit and obtain a linear model of the power converter. In order for our average value model to be valid, we must ensure that operation

is limited to the CCM regime. In CCM, the current in the inductor is always greater than zero, thus inductor stored energy is never depleted.

To obtain the duty cycle average value model, we time-average the differential Equations (2.1) and (2.2) over one duty cycle as in

$$di_{L-AVG} = \frac{1}{T}\int_0^T iL \cdot d\tau$$

$$di_{L-AVG} = \frac{1}{T}\int_0^{T_{on}} iL \cdot d\tau + \frac{1}{T}\int_{T_{on}}^{T_{off}} iL \cdot d\tau$$

$$di_{L-AVG} = \frac{1}{T}\left(\frac{E}{L} - \frac{V_C}{L}\right) \cdot T_{on} + \frac{1}{T}\left(-\frac{V_C}{L}\right)T_{off}$$

$$di_{L-AVG} = \left(\frac{E}{L} - \frac{V_C}{L}\right) \cdot D + \left(-\frac{V_C}{L}\right)(1-D)$$

$$dV_{C-AVG} = \left(\frac{i_L}{C} - \frac{V_C}{CR}\right) \cdot D + \left(\frac{i_L}{C} - \frac{V_C}{CR}\right)(1-D)$$

$$di_{L-AVG} = \left(\frac{E \cdot D}{L} - \frac{V_C}{L}\right)$$

$$dV_{C-AVG} = \left(\frac{i_L}{C} - \frac{V_C}{CR}\right)$$

.   (2.5)

When we examine the results of the average value analysis, we see that in steady-state, the capacitor voltage is equal to the duty cycle times the input voltage, and the inductor current is equal to the load current. This result is supported by the textbook analysis of [10]. The dynamics of the buck converter are exactly the same for the dynamics of a simple RLC circuit but with the input voltage modulated by the duty cycle. To complete our model, we next consider the current produced by the voltage source E. By a simple power balance, the power produced by the voltage source E ($P_E = EI_E$) must be equal to the power produced by the average value input voltage supply ($P_{ED} = EDi_L$); therefore, $I_E = Di_L$. The full average value circuit model is illustrated in Figure 4. Similar analysis of elementary boost and buck-boost converters yields similar average value models. These are presented for completeness in Figures 5 and 6. The main purpose of showing the average value models of these various converters is to demonstrate that, regardless of

10

the topology, elementary switching power converters can be reduced to simple second-order models. Even though these average value models are simple, they preserve the dynamics of the original system but without the complexity of switch-based models. The use of average value models has proven to be accurate and instructive in many applications while providing the benefits of using linear tools and much simpler and faster computer simulations.
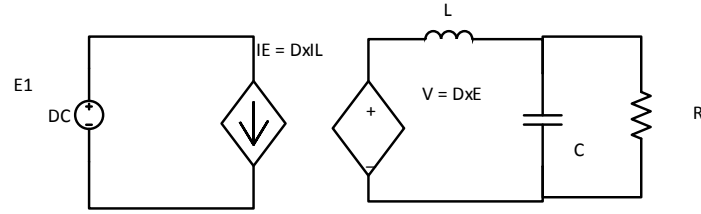
Figure 4. Buck Converter Average Value Model
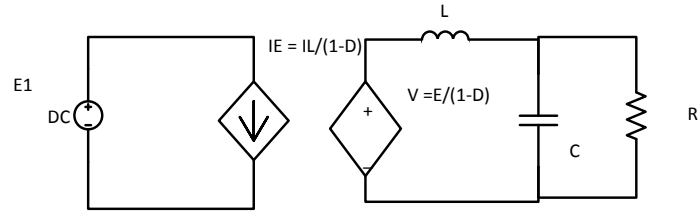
Figure 5. Boost Converter Average Value Model
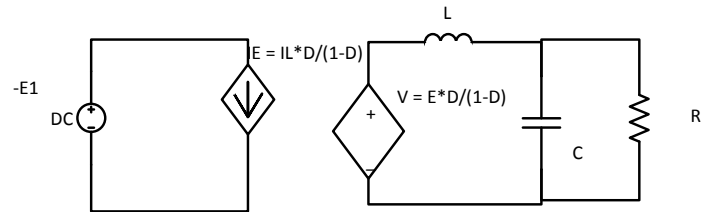
Figure 6. Buck-Boost Converter Average Value Model

D. CONSTANT POWER LOADS

After discussing power converters, we naturally segue into the concept of CPL. If a power converter is coupled to a control device, such as a regulator, then the voltage gain of the converter can be adjusted, via the duty cycle, to respond to changes in loading and input voltage. For example, in a laptop computer power supply, regardless of the voltage coming from the wall socket, the laptop power supply provides consistent high-quality DC power to the laptop computer. Any variation of the input voltage or frequency is counteracted by the regulator to ensure that the laptop receives the correct voltage. Since the voltage provided by the converter is constant and we assume that the load is also constant, the regulated converter consumes power at a constant level.

Naturally, as we have seen with our average value models, the power converter, even with a regulator, has transient dynamics; however, if the transient dynamics of the regulated converter are much faster than the transient dynamics of the rest of the power supply network it is connected to, then we may ignore the dynamics of the converter. This is the idealized CPL.

1. CPL Impedance

An idealized CPL is a tightly regulated power converter whose dynamics may be effectively ignored [11]. The idealized CPL presents problems for the design, analysis, and especially the stability of the systems of which they are a part. This is due to the non-linear negative impedance of the CPL. The simple Ohm's law relationship of the CPL is inscribed as

$$P_{CPL} = V_{CPL} I_{CPL}.$$

(2.6)

From Equation (2.6), it follows that the DC, or large-signal, resistance of the CPL is

$$R_{large-signal} = \frac{V_{CPL}}{I_{CPL}} = \frac{V^2_{CPL}}{P_{CPL}} = \frac{P_{CPL}}{I^2_{CPL}}.$$

(2.7)

The small-signal resistance of the CPL is obtained by taking the derivative of CPL voltage with respect to CPL current to obtain

$$R_{small-signal} = \frac{d}{dI_{CPL}}V_{CPL} = \frac{d}{dI_{CPL}}\frac{P_{CPL}}{I_{CPL}} = -\frac{P_{CPL}}{I^2_{CPL}} = -\frac{V^2_{CPL}}{P_{CPL}}.$$
(2.8)

From Equation (2.8), we clearly see the negative and non-linear nature of CPL impedance. The concepts of Equation (2.8) are demonstrated visually in the CPL I-V curve of Figure 7. In Figure 7, it is apparent that CPL impedance is strongly affected by changes to bus voltage. As we see in further analysis, the negative aspect of CPL impedance poses stability issues for any DC distribution system that supplies CPL. The negative impedance is not the most confounding aspect of CPL impedance; instead, it is the non-linearity.

Engineers have a wide variety of tools available to analyze and control linear systems. Time-domain and frequency-domain analysis tools are well-known and understood. Determining whether or not a system is stable is usually just a matter of determining the eigenvalues of the system matrix in the time domain or checking gain phase plots for sufficient margin in the frequency domain. With a non-linear system, stability analysis and control tools require more advanced training and study.
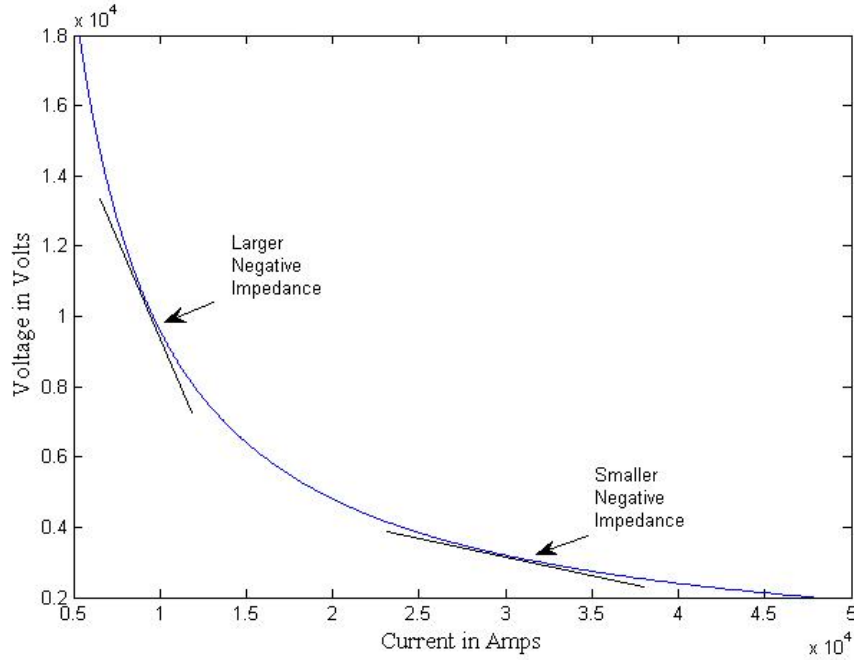
Figure 7. I-V Curve for 96 MW CPL

2. CPL Stability

Now that we have a model for CPL impedance, we can perform simple analyses to gain insight on how CPL affect DC distribution system stability. For purposes of illustration, we examine CPL stability in the context of a system of cascaded buck converters. We use the average value model of the buck converter developed previously, shown in Figure 8. For this relatively simple circuit we will first discuss stability in the context of a linearized system, and then we will briefly touch on non-linear Lyapunov stability analysis.
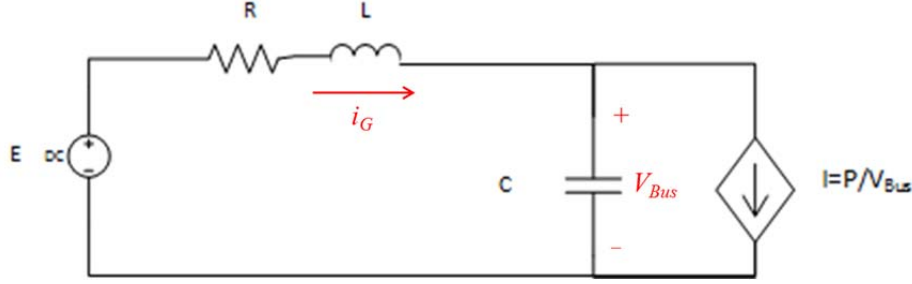
Figure 8. Average Value Model Buck Converter with CPL

a.   *Linearized System Analysis*

For the system of Figure 8, we first derive the differential equations from Kirchoff's voltage law and Kirchoff's current law to obtain

$$\dot{i}_G = \frac{-R}{L}i_G - \frac{1}{L}V_{Bus} + \frac{1}{L}E$$

$$\dot{V}_{Bus} = \frac{1}{C}i_G - \frac{1}{CR_{CPL}}V_{Bus}$$

,   (2.9)

where RCPL is the linearized CPL impedance and all other parameters are as previously defined. Next, we use the characteristic polynomial to establish stability conditions. We obtain the characteristic polynomial by first re-writing the differential equations in state-space form, then taking the determinate as in

$$\begin{bmatrix} \dot{i}_G \\ \dot{V}_{Bus} \end{bmatrix} = \begin{bmatrix} \dfrac{-R}{L} & \dfrac{-1}{L} \\ \dfrac{1}{C} & \dfrac{-1}{CR_{CPL}} \end{bmatrix} \begin{bmatrix} i_G \\ V_{Bus} \end{bmatrix} + \begin{bmatrix} \dfrac{1}{L} \\ 0 \end{bmatrix} E$$

$$\dot{x} = Ax + Bu$$

$$\det[\lambda I - A] = \lambda^2 + \lambda\left(\frac{R}{L} + \frac{1}{CR_{CPL}}\right) + \frac{1}{CL}\left(1 + \frac{R}{R_{CPL}}\right).$$ (2.10)

The purpose of this exercise is to gain insight into how changing certain circuit parameters will affect stability. The first thing we must recall is that R_{CPL} is negative. In order for the system described by Equation (2.10) to be stable, all of the terms must have the same sign. When examining the second term of the characteristic polynomial in

Equation (2.10), we see that increasing R, reducing L, increasing C, and increasing the magnitude of R_{CPL} all improve stability. In regard to these methods, increasing R is undesirable, since this reduces the steady-state power efficiency of the system. The inductance L is usually reduced to minimum values anyway since inductors have parasitic resistance and are generally large components. As we saw in part B of this chapter, there is a limit to how small L can be; it must be equal to or larger than the critical inductance. Increasing the magnitude of R_{CPL} can be achieved by raising system voltage or by reducing load power. This leaves increasing bus capacitance as the free design variable for improving stability margins. Depending on the voltage and power of the system, we find that bus capacitance levels can be quite large, with values in the tens of mF.

Despite all of this analysis, we must recall that this is only a linearization of the system. This analysis is only valid for small deviations about the nominal V_{Bus} voltage. Any large deviation, such as one that might be induced by a large load transient or a fault, could violate our assumptions and result in operation within an unstable region.

b.   *Lyapunov Analysis*

Lyapunov analysis techniques are required to assess large-signal regions of attraction (ROA). We follow the stability analysis methods of [12] to obtain the necessary conditions for the circuit of Figure 8. First, we must transform the first order differential equations of Equation (2.9) by a change of variables where (V_0, I_0) represents the desired equilibrium operating point. We define the new state variables as Z_1 and Z_2 according to

$$
\begin{aligned}
Z_1 &= V_{Bus} - V_0 \\
Z_2 &= i_G - I_0
\end{aligned}
\tag{2.11}
$$

We substitute the new state variables from Equation (2.11) into Equation (2.9) to obtain

$$
\begin{aligned}
\frac{dZ_2}{dt} &= -\frac{R}{L}(Z_2 + I_o) - \frac{1}{L}(Z_1 + V_0) + \frac{E}{L} \\
\frac{dZ_1}{dt} &= \frac{1}{C}(Z_2 + I_0) - \frac{P}{C(Z_1 + V_0)}
\end{aligned}
\tag{2.12}
$$

If we assume $E = RI_0 + V_0$, Equation (2.12) can be rewritten as

$$\ddot{Z}_1 + \frac{1}{LC}\left(RC - \frac{LP}{(Z_1 + V_0)^2}\right)\dot{Z}_1 + \frac{1}{LC}\left(Z_1 - \frac{(RI_0 Z_1)}{(Z_1 + V_0)}\right) = 0 \qquad (2.13)$$

Since Equation (2.13) is a second-order equation of the form of

$$\ddot{Z}_1 + a \cdot f(Z_1)\dot{Z}_1 + b \cdot g(Z_1) = 0 \qquad (2.14)$$

a Lyapunov function, ψ, can be found according to

$$\psi = b \cdot \int_0^{Z_1} g(\vartheta)\,d\vartheta + \frac{\dot{Z}_1^2}{2}$$

$$\dot{\psi} = -a \cdot f(Z_1)\dot{Z}_1^2 \qquad (2.15)$$

The resulting Lyapunov function and Lyapunov derivative are

$$\psi = \left(\frac{1}{CbusLg1}\right)\left\{\frac{Z_1^2}{2} - Rg1I_0 Z_1 - Rg1I_0 V_0 \log\left(\frac{Z_1 + V_0}{V_0}\right)\right\} + \left(\frac{1}{2Cbus^2}\right)\left\{Z_2 + \frac{I_0 Z_1}{Z_1 + V_0}\right\}^2 \qquad (2.16)$$

and

$$\dot{\psi} = \left(-\frac{1}{Lg1Cbus}\right)\left(Rg1Cbus - \frac{Lg1P}{(Z_1 + V_0)^2}\right)\dot{Z}_1^2 \qquad (2.17)$$

respectively. For an initial condition (v,i) to be within the ROA centered about the equilibrium point (V_0, I_0), the Lyapunov function, Equation (2.16), must be positive definite, and its derivative, Equation (2.17), must be negative definite. We substitute Equation (2.11) back into Equations (2.16) and (2.17) and obtain the inequalities

$$\frac{L_{g1}}{2Cbus}\left\{i - I_0 + I_0 * \frac{v-V_0}{v}\right\}^2 \geq \left\{-\frac{1}{2}(v - V_0)^2 + R_{g1} * I_0 \left[v - V_0 - V_0 \ln\left(\frac{v}{V_0}\right)\right]\right\} \quad (2.18)$$

and

$$v \geq sqrt\left(L_{g1}\frac{P}{R_{g1}C_{bus}}\right). \quad (2.19)$$

The inequalities of Equations (2.18) and (2.19) determine the ROA for the system and chosen input. Initial conditions within the ROA eventually settle back to equilibrium conditions. Those initial conditions that do not satisfy Equations (2.18) and (2.19) result in oscillatory or unstable system dynamics. An example ROA is presented in Figure 9 for the system of Figure 8. Capacitance is varied to show the effects bus capacitance has on the ROA. The region displayed is limited to a region from zero volts to double the steady-state voltage and from zero amperes to double the steady-state value. From Figure 9, we see that the ROA is an ellipsoid centered about the equilibrium point whose major axis is oriented somewhat parallel to the current axis and whose minor axis is parallel to the voltage axis. By inspection of this ROA, we can conclude that the system is far more sensitive to changes in voltage than to changes in current. Furthermore, small changes in capacitance can have enormous consequences for the size of the ROA. While this analysis is valid for the circuit of Figure 8, conducting non-linear analysis for high-order systems can be quite complicated. Finding appropriate Lyapunov functions is more art than science. While some Lyapunov functions are known to be appropriate for certain systems, any change of the dynamics of that system can render the Lyapunov function invalid.
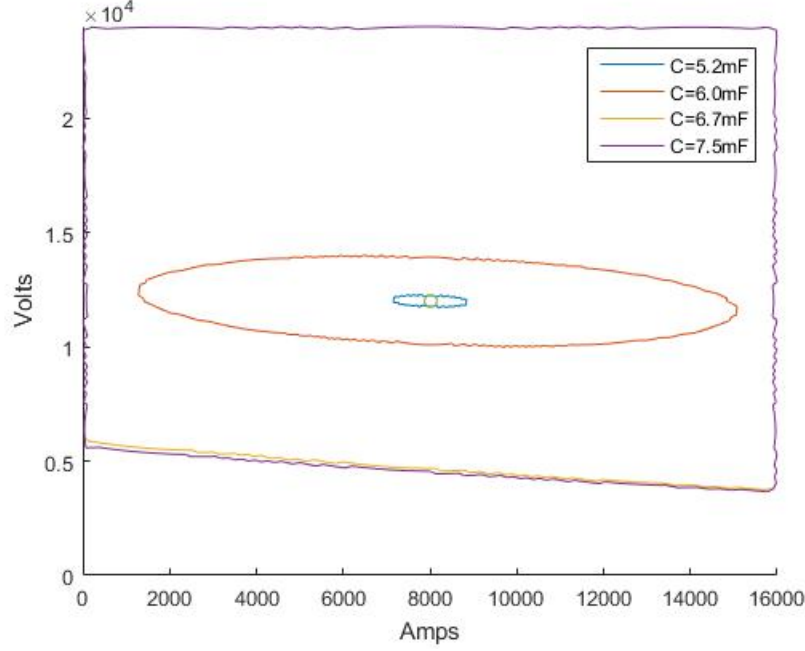
Figure 9. ROA for a 12 kV Second Order System with 96-MW CPL

E.   RELATED WORK

A great number of authors have applied control techniques to expand the ROA in DC power systems with CPL. The controls applied to this problem include a cornucopia of techniques. For ease of understanding, we summarize the control techniques applied to this problem into two main groups: linear and non-linear.

1.   Linear Methods

Linear methods are those methods developed for control of linear systems.

a.   *Passive Damping*

Passive damping techniques involve linearizing the CPL about an expected equilibrium point. From this equilibrium point, Equation (2.10) can be used to select appropriate component values for the desired performance. This method can be reduced to minimizing inductance, using large capacitors, placing large resistances in parallel with the CPL, or placing a damper resistance in line with the capacitor as suggested by Hodge, Flower, and Macalinden [11].

b.   *Active Damping*

Active methods improve dynamics by including some variant of a proportional, integral, and derivative (PID) controller to provide the desired dynamics. These methods are sometimes dubbed "virtual impedance" methods. Magne et al., in [13], apply the control signal to the CPL regulator to introduce a "virtual capacitor." A passivity based criteria is used to develop the PID gains in [14]. Zadeh et al. use standard PID control applied to both a current control loop and a voltage control loop in [15] and [16] to stabilize system voltage. Grillo et al. [17] apply proportional current and voltage gains for his controller.

The approach of Carmeli et al. in [18] is interesting in that rather than controlling the supply voltage or the CPL controller response, they chose to place an energy storage device (ESD) in parallel with the CPL. They then use a PI controller on the ESD to source or sink current to enhance the stability of the system. In the conclusions, [18] notes that compared to a bulk capacitor, the ESD provides a better dynamic response as well as providing the opportunity to store the energy at lower voltage than the main bus voltage. This is especially important for MVDC as capacitors rated in the 10 kV–20 kV range are heavier, bulkier, costlier, and more failure prone than equivalent capacitors rated at lower voltages.

2.   **Non-linear Methods**

a.   *Adaptive State-Feedback*

Arcidiacono et al. [19] demonstrate use of an adaptive proportional state-feedback controller to ensure stable operation throughout the full range of operation for a second-order system. The proposed controller adaptively updates proportional voltage and current feedback gain to control the duty cycle of a supply DC-DC converter in order to provide stable and consistent regulation.

b.   *Feedback Linearization*

Feedback linearization is a popular technique that involves a two-part control process. In the first part of the control process, whenever possible the state of the system

is properly redefined so that the resulting system is linear. The second part of the process involves using traditional linear pole-placement techniques, such as PID controllers, to obtain the desired system behavior. Ciezki and Ashton [20] applied this technique to a buck converter with an attached CPL in the late 1990s. Sulligoi et al. developed feedback-linearization controller implementations in [21] and [22] for multi-machine multi-CPL systems. Their approach reduced a multi-machine problem to a second-order single-input representation. It was further demonstrated in [23] that feedback linearization can provide stability under controller saturation conditions. In [24], active damping control is effectively used for the generating DC-DC converter controller while feedback linearization is used for main bus-connected buck converters serving CPL. This method was proved to be far superior to active damping when disturbances were large [25].

c. *Linear Quadratic Gaussian*

Linear Quadratic Gaussian (LQG) control is a technique where there is uncertainty in the state of the system due to white Gaussian noise and modeling errors. The state variables are estimated using a Kalman filter (EKF) and a linear quadratic regulator (LQR) is then used to provide state-feedback gains that minimize a quadratic cost function. Zhu, Liu, and Cupelli used this technique in [26] to provide decentralized control of a multi-machine multi-load hypothetical MVDC shipboard system with CPL. By using the EKF, the authors were able to represent each generating DC-DC converter as having a single CPL, thus reducing the order of the problem to 2nd order. This technique does not require the significant number of simplifying assumptions needed in [22] to reduce a complex system to 2nd order. Load sharing between machines is enforced by proportioning the estimated load calculated by the EKF. In this way, robustness on loss of generating power is accomplished by sensing the loss and re-apportioning load to the remaining generators. Performance of LQG was validated in a hardware-in-the-loop simulation [27].

Neither [26] nor [27] describe the construction of the EKF or the *Q* and *R* matrices which describe model and observation uncertainties required for LQG implementation. This omission confounds this author's ability to reproduce their work.

d.   *Lyapunov Backstepping*

Backstepping is a technique where a non-linear control law is developed using a chosen Lyapunov function. Since CPL power levels are variable in real-life applications, these values must be estimated. Adaptive backstepping introduces estimator variables as well as other design variables necessary to ensure Lyapunov stability conditions. Like [22] and [26], Cupelli et al. [28] again reduce the system to second order so that a known Lyapunov function can be used. Backstepping produces controllers which outperform LQG in terms of overshoot, undershoot, and settling time response as shown in both [28] and [29]. Unfortunately, the claimed performances depend upon parameter choices which are not well-defined. As mentioned previously, this method is only applicable in cases where the system Lyapunov function is known.

e.   *Synergetic Control*

Synergetic control is a method where, after deriving the state space model, macro variables are defined from which control laws are calculated. This method was applied by Kondratiev and Dougal [30], [31]. Cupelli et al. [32] found synergetic control performance superior to feedback linearization; however, synergetic control is much more difficult to grasp than previously mentioned methods and has not gained popularity.

F.   SUMMARY

In this chapter, we explored the concept of the Electric Warship. To make the Electric Warship a possibility, the U.S. Navy is pursuing the development of MVDC distribution systems. These distribution systems rely on the performance of a wide variety of electronic power converters.

Switch-based power electronic devices are inherently non-linear; however, in systems where the device is switched according to a duty cycle over a defined period, linear average value models of the power converters may be developed. Linear average-

value models simplify the analysis and simulation of power-electronic converters and provide a familiar entry point for developing control strategies.

The use of power converters that are tightly regulated by high bandwidth controllers result in CPL. CPL present a non-linear negative impedance characteristic, which complicates stability analysis and control strategies. While a system may be stable at an equilibrium point, the system itself is not globally stable. Unlike a linear system that is either stable or unstable, non-linear systems are only stable within an ROA about the equilibrium point. Further complicating matters is that Lyapunov analysis for high-order systems can be very difficult or impossible, since finding appropriate Lyapunov functions is more art than science.

Various control strategies have been employed to improve ROAs for systems with CPL. The earliest methods explored were passive damping methods, which add additional resistance and capacitance to the system. Passive damping methods are simple and reliable, but they introduce weight, cost, and inefficiencies. Linear control methods employ traditional PID controllers to expand the ROA. Linear control methods are more complicated than simple passive damping but utilize the massive body of knowledge available to engineers for control of linear systems. Unfortunately, the system is non-linear, so global stability is not guaranteed.

Non-linear control methods have also been employed. Some non-linear methods, such as adaptive linearization and state-feedback linearization, are fairly straightforward and user-friendly. This is due to their close similarity to linear methods. Strict non-linear methods, such as adaptive backstepping and synergetic controls, are available but have found little popularity due to complexity and ambiguity in developing control laws.

With the exception of the obtuse synergetic control, all of the control schemes surveyed are single-input schemes. Control schemes in [15], [19]–[21] approach multi-machine systems by either simplifying multiple co-located input devices into a single device or by decomposing the system into multiple independent systems with a single-input. The approach of simplifying co-located input devices into a single input device is inadequate for systems with distributed input devices. Neither does that approach allow

23

input devices that are significantly dissimilar. The second approach, used by the LQG controllers, decomposes the multi-input system into several independent single-input mini-systems. This approach works well for distributed systems and where dissimilar input devices are used. A disadvantage of that approach is that controllers act independently to regulate system performance. The decentralized and uncoordinated nature of this approach introduces the possibility that the separate controllers may contradict one another, resulting in suboptimal regulation.

III. NAVSEA CIRCUIT MODEL

Before we explore an appropriate controller for a shipboard electrical distribution system, we first need a model for it. The starting point for this analysis is a proposed architecture for a next generation Electric Warship from [33]. In this chapter, we walk through the development of an experimental circuit model. In the first section, we describe the key elements of the distribution system. Next, the NAVSEA proposed distribution system is simplified into an abbreviated system. Finally, an equivalent circuit structure is developed.

A.   NAVSEA PROPOSED DISTRIBUTION SYSTEM

A proposed naval electrical distribution system, described in [8] and illustrated in Figure 10 provided by Dr. Norbert Doerry of NAVSEA, has a MVDC main distribution system. While several voltage ratings are possible, 12.0 kV appears to be the most likely candidate.

The concept includes four power generation modules (PGMs) which can be either gas turbine or diesel engine prime movers driving split-winding multi-phase AC generators. The output of the AC generators are rectified and used as the supply voltage for a DC-DC converter. In the proposed NAVSEA system, there are two large PGMs, called Main Turbine Generators (MTG), and two smaller PGMs, called Auxiliary Turbine Generators (ATG). The four PGMs are connected to Port and Starboard distribution buses.

The distribution buses span several zones, each zone representing a watertight boundary or functional grouping within the ship. The zones contain switchboards, intermediate DC-DC power converters, ESD, and point-of-load DC-DC and DC-AC power converters. The point-of-load power converters transform the intermediate DC power from the intermediate DC-DC converters to low-voltage DC and the various forms of AC power used within a ship: 120 V/240 V/450 V 60 Hz and 120 V 400 Hz.

The load types can be divided into three main groups. The first group is referred to as Ship's Service loads. These are loads such as lighting, air conditioning, wall

sockets, galley equipment, and other various small loads. Ship's Service loading is generally considered constant.
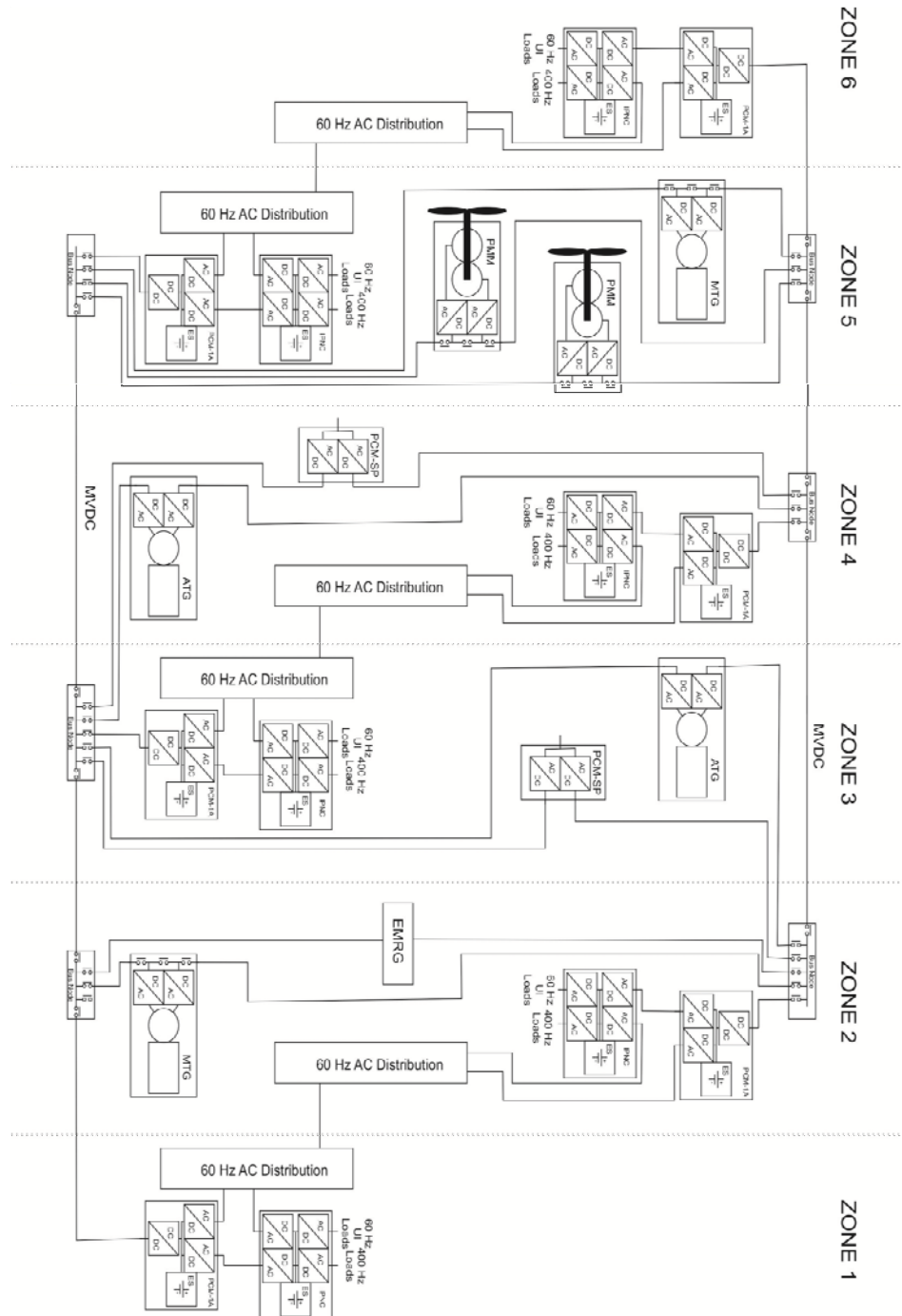
Unpublished diagram provided by Dr. Norbert Doerry of NAVSEA

Figure 10. NAVSEA Proposed Electrical Distribution System.

The next type of load is referred to as Combat System loads. These loads are associated with energy weapon systems. Whether the load is a high-power radar, laser weapon system, or naval railgun, these loads are characterized as pulsed or stochastic type loads. Combat System loads can experience large step-changes in power consumption. A free-electron laser, for example, can consume power on the order of 10 MW during discharge but consumes much less power during standby conditions. Likewise, a naval electromagnetic railgun (EMRG) consumes a relatively small amount of power in standby compared to the period when it is charging its energy storage system. An EMRG operating at maximum cyclic rate of fire could require 25 MW or more [34].

The final class of load is propulsion motors. Propulsion loads have historically been the largest loads on a ship. The U.S. Navy Zumwalt (DDG-1000) class destroyer is a 15000+ ton warship with two advanced induction propulsion motors rated at 35.0 MW each for a total of 70.0 MW of propulsion capacity [35].

B.   SIMPLIFIED DISTRIBUTION SYSTEM

To aid in producing an experimental circuit model, we reduce the diagram shown in Figure 10 into a simplified model. The simplified distribution system should retain the key features shown in Figure 10. The first key feature that we notice is the multiple PGMs. These PGMs come in two capacities: MTG and ATG.

The second feature we extract is multiple power conversion layers. The first level of power conversion is from the PGM to the main DC buses. The second level is from the main DC buses to the zonal DC buses via the intermediate DC-DC converters. Third level and greater power conversion layers occur via point-of-load converters within the zones. Since we wish to focus our attention on stability and control of the MVDC main buses, we simplify all third-level and beyond power converters by representing them as ideal constant power loads. This simplification is reasonable since these converters operate at lower voltages and can be cost-effectively switched at very high speeds (i.e., $f_{switch} > 20$ kHz). If we make the additional assumption that the point-of-load converters are tightly regulated, we have the necessary conditions to ascribe CPL behavior to the third-level and greater power converters.

The third feature we extract from Figure 10 is the zonal architecture. We construct our model with three zones based on the three types of loads present. Zone #1 contains Ship's Service loads, which make up a relatively small fraction of overall loading (<20%) and have relatively small load variation. Zone #2 contains Combat System loads, which are pulse-type loads, capable of huge jumps in power in a matter of microseconds (up to 25-MW pulses). Zone #3 holds the propulsion loads, which are the largest loads on the ship. Propulsion loading can be either static or highly variable depending on the time-scale. Maneuvers from all-back full to ahead flank, as well as sea-swell periods, are on the order of seconds. In order to test worst-case conditions, we assume that all loads step instantaneously.

The final feature we extract is the placement of ESD. When we look carefully at Figure 10, we notice that every intermediate DC-DC converter has a HESS unit associated with it; therefore, the Ship's Service and Combat System zones have an ESD connected to their intermediate buses. Propulsion, however, does not have an associated HESS, since the ship's inertia is sufficient for stored energy.

The culmination of our simplifications is illustrated in the block diagram of Figure 11. Figure 11 has all of the key features. The four PGMs are two different sizes. The two larger PGMs are 40.0-MW capacity while the two smaller PGMs are 10.0-MW capacity for a total generating capacity of 100.0 MW. The MVDC bus is nominally set at 12.0 kV DC. There are three load zones. The Ship's Service intermediate DC-DC converter transfers the 12.0-kV MVDC bus power to a 1.0-kV DC Ship's Service bus. The Ship's Service zone is a 20.0-MW maximum load CPL with a bus-connected ESD. The Combat System intermediate DC-DC converter transfers MVDC bus power to a 6.0-kV bus. The Combat System zone has a 30.0-MW maximum capacity CPL with a bus-connected ESD. The Propulsion intermediate converter transfers 12.0-kV MVDC power to 10.0 kV DC for an 80.0-MW maximum capacity CPL. It is important to note, that although maximum possible loading exceeds maximum available generating power, exercising the plant through overload conditions is beyond the scope of this document.

A harmonic filtering element is included at the input to the intermediate DC-DC converters since such filters are often necessary in practical applications.
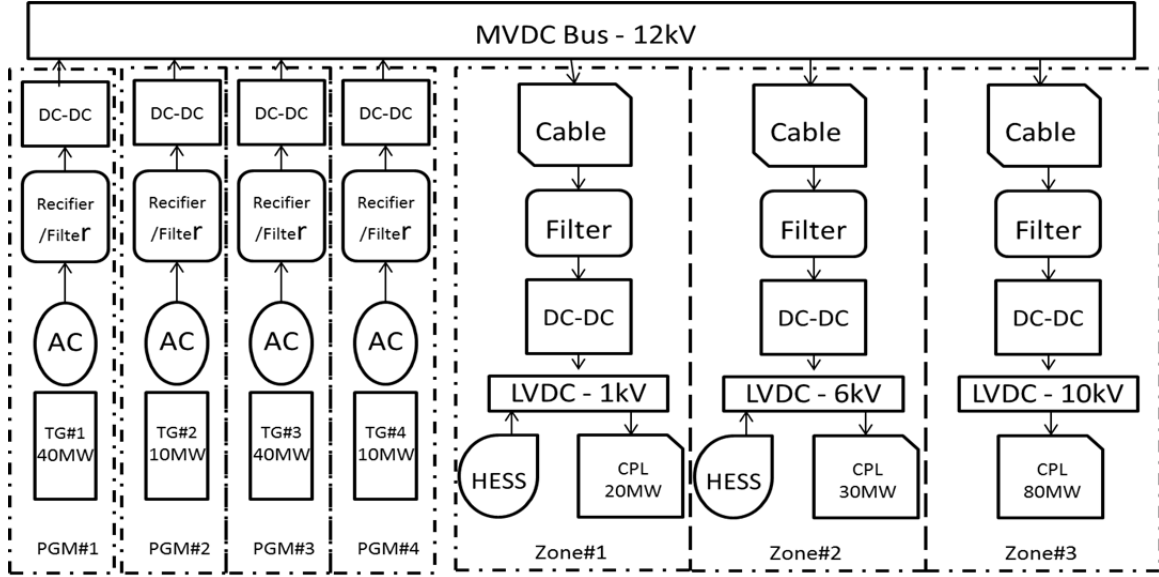
Figure 11. Simplified Distribution System Model

C.   EXPERIMENTAL CIRCUIT MODEL

Transforming the distribution system model of Figure 11 into a circuit model is our next task. We systematically move from the PGMs to the loads in assigning a circuit model to each block in the diagram.

PGMs are the first blocks we tackle. The PGMs are composed of a prime mover, multi-phase AC generator, rectifier, and DC-DC converter. In order to make our problem tractable and relatively easy to simulate, we assume that the rectified AC from the generator is an ideal DC source. While not strictly true, it allows us to ignore motor and generator rotational dynamics. This is a desirable assumption, since developing a more realistic model for the generator would be quite time-consuming and ultimately have little to no bearing on the results of this research. This assumption also allows us to assign the PGM the form of the DC-DC converters in Chapter II, Figures 4–7. We can further simplify the circuit model for the PGM if we simply ignore the left-hand side of Figures 4–7. This reduces the PGM circuit model into a controlled voltage source, a series RL impedance, and a parallel filter capacitor. This way, we do not need to know what type of DC-DC converter is being used for the PGM DC-DC converter. Since the PGMs are

operating at a fairly high voltage (12.0 kV), we assume a DC-DC converter switching speed of 1.0 kHz. This speed is easily possible with existing semiconductor technology.

Next, we examine the bus cabling. Using the short-wire impedance model for the MVDC bus, we represent MVDC bus cabling as a series RL impedance with a parallel capacitance. This model has basis in other work, as seen in [27]. For the sake of complicating the model such that the simplifying assumptions of previous controllers cannot be used, we assume that bus work represents 300 meters of cabling. Cabling is then paralleled to meet the MVDC current demands of the connected zone. We choose 300 meters since that distance represents the fore-aft length of an aircraft carrier, the U.S. Navy's largest platform. We consider this to be a worst-case assumption, since Figure 10 shows that PGMs and intermediate DC-DC converters are distributed throughout the ship.

The input filter is selected to be a series-RC filter connected from the input of the intermediate DC-DC converter to ground. This filter was selected because it adds a stabilizing pole to the circuit.

The intermediate DC-DC converter circuit models are identical to Figure 4. We chose a fixed duty cycle for the intermediate DC-DC converters, as this effectively renders them as "DC transformers." This choice prevents us from simplifying the intermediate DC-DC converters into ideal CPL as other authors have done. The critical inductance and minimum capacitance for each intermediate DC-DC converter are calculated assuming a 1.0-kHz switching speed.

Finally, we consider the point-of-load converters and ESD. The point-of-load converters are assumed to be ideal CPL. This assumption seems appropriate since the loads operating on the zone buses operate at lower voltages than the converters connected directly to the MVDC bus. This allows the point-of-load converters to operate at much higher switching frequencies and exhibit CPL behavior. The ESD are assumed to be HESS with both fast dynamics and high power capacity. In order to avoid modeling another complex system, the HESS is assumed to behave as a controlled current source as has been done by previous authors. We assume that HESS are connected to the zone

buses by switching power supplies. Just as with the CPL, we assume that the lower voltage on the zone buses allows faster switching speeds. Here we have chosen 16.0-kHz switching speeds for the HESS. This number was selected because it is significantly faster than the PGM at 1.0 kHz and because the switching period divides into the PGM switching period by a convenient integer value.

The complete experimental circuit model is shown in Figure 12. Circuit component names are in black lettering, while state-variables are written in red with direction arrows showing the positive flow of currents and +/− symbols indicating polarity for voltages. The R_{gx} and L_{gx} values indicate the PGM output impedances. Since all four PGMs are in parallel, the PGM output capacitors have been combined into one equivalent bus capacitor, C_{bus}. The R_{zx}, L_{zx}, and C_{zx} values indicate the short-wire impedance values. The parameters R_{dx} and C_{dx} are for the damper filter components. The parameters D_1, D_2, and D_3 are the CCM calculated duty cycles of the intermediate buck converters. The duty cycle values relate the buck converter input current to the buck converter output voltage. The parameters L_{bx} and C_{bx} are the inductances and capacitances, respectively, for each of the intermediate buck converters. The controlled current sources with downward pointing arrows are the point-of-load CPL, while the currents sources with upward pointing arrows are the HESS.

D.   SUMMARY

In this chapter, we stepped through the process of converting a proposed NAVSEA electric warship distribution system into an experimental circuit model. The key features of the NAVSEA model were extracted and used to make a simplified block diagram. From the simplified block diagram, circuit topologies and properties were assigned. The final topology consists of four average-value model DC-DC converters connected to a 12.0-kV MVDC bus. Also connected to the MVDC bus are three load zones. Each load zone has cable impedance, an input RC filter, and an average value buck converter operating in CCM at fixed duty cycle. On the intermediate buses, all loads were assumed to be CPL, and the bus-connected HESS was modeled as an ideal controlled current source.

31
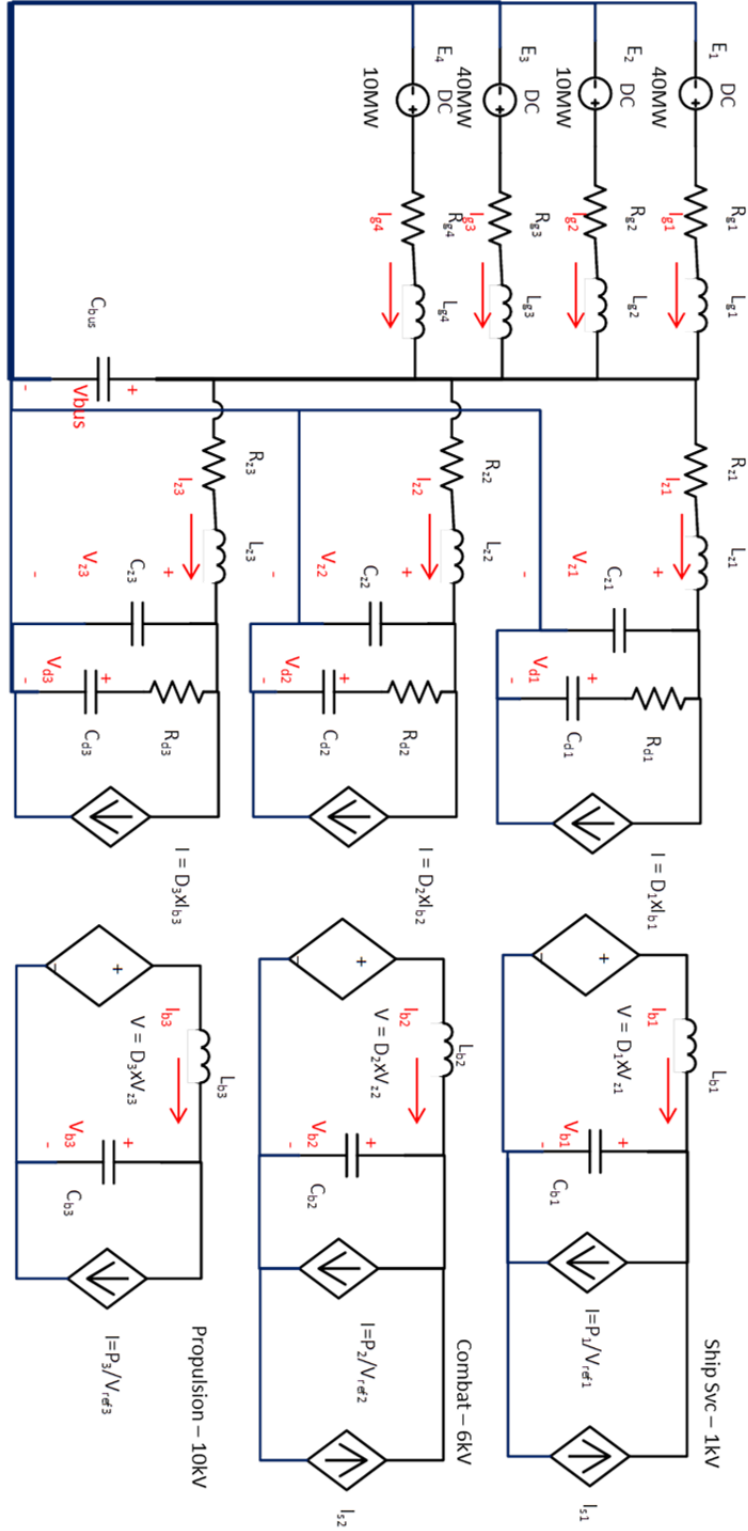
Figure 12. Experiment Circuit Model

IV. ADAPTIVE, MULTI-RATE LINEAR QUADRATIC REGULATOR

A. LINEAR QUADRATIC REGULATOR

The source document for this section is the Donald Kirk text [36]. LQR is an optimal control technique for linear systems. One of its attractive features is that it can be implemented recursively in an efficient manner using dynamic programming. The plant is described in state-space form by the system of first order linear differential equations as in

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \tag{4.1}$$

with A representing the plant's state matrix, B representing the feedback matrix, x representing the state-vector, and u representing the input vector. The performance measure to be minimized, J, is represented by the integral equation

$$J = \frac{1}{2}x^T(t_f)Hx(t_f) + \frac{1}{2}\int_{t_0}^{t_f}\left[x^T(t)Q(t)x(t) + u^T(t)R(t)u(t)\right]dt \tag{4.2}$$

where H is the boundary cost matrix, Q represents the state-error cost matrix, and R represents the input cost matrix. The matrices H and Q must be real, symmetric, and positive semi-definite. The matrix R must be real, symmetric and positive definite. This is to ensure that the inverse to R exists. For a problem with finite time duration ($t_f < \infty$), the optimal control is found by recursively solving the Riccati equation,

$$\dot{K}(t) = -K(t)A(t) - A^T(t)K(t) - Q(t) + K(t)B(t)R^{-1}(t)K(t), \tag{4.3}$$

for $\dot{K}(t)$ starting at t_f with $K(t_f) = H$ and then working backward in time until t_0, with the optimal input $u(t)$ computed by

$$u(t) = -R^{-1}(t)B^T(t)K(t)x(t). \tag{4.4}$$

In the special case where certain conditions are met, Equations (4.2)-(4.4) can be dramatically simplified while still preserving optimal state trajectories. The special

conditions are: 1. the plant is completely controllable; 2. A, B, Q, and R are constant matrices; 3. $H = 0$; and 4. $t_f = \infty$. Specifically, in the limit of t_f approaching infinity, $K(t)$ becomes a constant matrix. For this special case, called infinite-horizon LQR, Equation (4.2) simplifies to

$$J = \frac{1}{2}\int_{t_0}^{\infty}\left[x^T(t)Qx(t)+u^T(t)Ru(t)\right]dt \quad , \tag{4.5}$$

while Equation (4.3) becomes

$$0 = -KA - A^T K - Q + KBR^{-1}K \quad , \tag{4.6}$$

and Equation (4.4 is rewritten as

$$u(t) = -R^{-1}B^T Kx(t) \quad . \tag{4.7}$$

While the completely controllable case provides optimal trajectories, LQR can still be used to stabilize the controllable modes in plants that are not fully controllable. In this vein, we expand the class of problems for which LQR is applicable to stabilizable systems.

B.   MULTI-RATE LQR

The multi-rate LQR controller theory presented in this section is derived from [37]. Analysis and design of multi-rate LQR controllers begins with the assumption that the system can be modeled as linear and time invariant (LTI) over a short enough time-period T. Over this short time-period, we may model the plant with a series of difference equations as in

$$x[k+1] = F[k]x[k] + G[k]u[k] \text{ for k=1,2,3...} , \tag{4.8}$$

where x denotes the state vector, F is a constant matrix that denotes the state matrix, G is a constant matrix that denotes the feedback matrix, u is the input vector and k denotes the time index. If the system is periodic over a sequence of L time steps, then a block-cyclic matrix equation of the form of

$$Z[i+1] = AZ[i] + BU[i], \qquad (4.9)$$

where

$$Z[i] = [x_1, x_L, x_{L-1}, \ldots, x_2]^T \quad U[i] = [u_1, u_L, u_{L-1}, \ldots, u_2]^T$$

$$A = \begin{bmatrix} 0 & F_L & 0 & 0 & \cdots & 0 \\ 0 & 0 & F_{L-1} & 0 & \cdots & 0 \\ 0 & 0 & 0 & F_{L-2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & F_2 \\ F_1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & G_L & 0 & 0 & \cdots & 0 \\ 0 & 0 & G_{L-1} & 0 & \cdots & 0 \\ 0 & 0 & 0 & G_{L-2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & G_2 \\ G_1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

characterizes the behavior of the system. For Equation (4.9), the vector Z is the super-state vector, A is the super-state matrix, B is the super-feedback matrix, and U is the super-input vector. Controllability and stabilizability of the L-step block-cyclic system may be found by the classical methods.

Under this L-cyclic regime, we see that it is possible to incorporate time-varying components such as $G[k]$ in a plant with inputs operating at multiple switching frequencies. In such a plant where the switching events are synchronized and the switching rates of the various inputs may be related by integer numbers of a common time-step, the input matrix $G[k]$ for this system is periodic and this method may be employed.

Computing block-cyclic LQR inputs is now quite straight-forward. Following the method of Part A, we may use the block-cyclic A and B matrices from Equation (4.9) with the block-cyclic $\hat{Q}$ and $\hat{R}$ matrices given as

$$\hat{Q} = \begin{bmatrix} Q_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & Q_L & 0 & 0 & \cdots & 0 \\ 0 & 0 & Q_{L-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & Q_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_2 \end{bmatrix} \quad \hat{R} = \begin{bmatrix} R_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & R_L & 0 & 0 & \cdots & 0 \\ 0 & 0 & R_{L-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & R_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & R_2 \end{bmatrix}, \qquad (4.10)$$

to compute the block-cyclic K matrix using the discrete-time Riccati equation,

$$0 = A^T KA + \hat{Q} - A^T KB(\hat{R} + B^T KB)^{-1}(A^T KB)^T \quad . \tag{4.11}$$

The solution to the Riccati equation yields a block-cyclic K whose components K_1-K_L are given as

$$K = \begin{bmatrix} K_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & K_L & 0 & 0 & \cdots & 0 \\ 0 & 0 & K_{L-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & K_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & K_2 \end{bmatrix} \quad . \tag{4.12}$$

The block-cyclic gains matrix is obtained by solving the block-cyclic version of Equation (4.7) given by

$$G = -(\hat{R} + B^T KB)^{-1} B^T KA \tag{4.13}$$

where

$$G = \begin{bmatrix} G_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & G_L & 0 & 0 & \cdots & 0 \\ 0 & 0 & G_{L-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & G_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & G_2 \end{bmatrix}$$

The input vector u calculated by the multiplication of the state-vector x by the appropriate sub-cycle gain matrix Gi of the sequence as in

$$u[k] = G_i x[k] \text{ for } i = 1, 2, \ldots, L. \tag{4.14}$$

Once the super gain matrix G and all of the G_i matrices are found, the input vector u for time index is found by cycling through the sequence of G_i matrices and multiplying them by the state-vector as in Equation (4.14).

This method, which we refer to as Periodic-Discrete LQR (LQR-PD), has a solid theoretical foundation and provides great value, especially when determining controllability. LQR-PD is quite computationally costly, especially for high order systems or systems with a large *L* number of sub-cycles in the sequence. In addition to requiring significant computation resources, we found that the MATLAB Riccati solver *dare()* had difficulty resolving large-order systems. The *dare()* function issued warnings regarding "ill-conditioned matrices" and "eigenvalues too-close to the origin." Furthermore, for some large systems, *dare()* could not resolve any solution at all.

Another drawback of LQR-PD pertains to dynamic systems. When the state matrix *A* changes, such as during a power transient in an electrical distribution system, LQR-PD is limited; it can only make adjustments once every super-cycle (*L* indices). Depending on the period of the super-cycle, LQR-PD may not be able to adequately react to dynamic events.

C.   SELECT MATRIX LQR

Select-Matrix LQR (LQR-SM) is very similar to LQR-PD except that input gain matrices are calculated at every sub-cycle using sub-cycle *A*, *B*, *Q*, and *R* matrices. Calculating in this fashion provides nearly equivalent results to LQR-PD while producing advantages in computational complexity and microcontroller memory requirements. LQR-SM as an adaptive, multi-rate controller for high-order non-linear systems was first presented in [38].

1.   Description

To overcome the limitations of LQR-PD, we need only make two minor changes to the computation routine. First, rather than compute all of the super-cycle gains using very large matrices as LQR-PD does, we simply shift to computing sub-cycle gains using the *A*, *B*, *Q*, and *R* matrices for the relevant sub-cycle. Second, rather than cycle through several *B* matrices, we choose to maintain a common *B* matrix and use sub-cycle specific *R* matrices. Calculating gains on the sub-cycle level improves the speed of calculation due to the Riccati solver (MATLAB *care()* or *dare()*) handling much smaller matrices.

This improvement in calculation speed is quite important when feedback gains must adjust to changes to the state matrix A in real-time, such as dynamic loading of a shipboard electrical distribution system.

R matrices are required to be symmetric positive definite; to meet this requirement while minimizing the number of design variables, we choose a diagonal R matrix. This choice both simplifies the number of design variables that must be selected by the designer but also simplifies the relationship between R matrix values and regulator output. In this way, the cost is a function of $u_1{}^2, u_2{}^2$, etc. and not of cross terms.

In order to effect the correct multi-rate response from the regulator, each separate sub-cycle has an associated R matrix. Those inputs that are changing value during the switching sub-cycle are assigned a diagonal R matrix value selected by the designer, while those inputs that are not changing during the sub-cycle are assigned a very large R matrix value. For this large R-value, we choose a value 100–1000 times larger than the largest value assigned to inputs that are changing during the sub-cycle. With this large penalty, the Riccati equation maximizes use of the unrestricted inputs in order to produce the optimal state trajectory while minimizing use of the restricted inputs. Practice has shown that the restricted input values calculated in this way are often so small as to be nearly zero. In reality, the restricted input value is zero. Since the calculated value of the restricted input is a close approximation to the actual value of the input, there is no great deviation from reality, and the integrity of our system is preserved.

To further generalize the application, we consider a non-linear system. The analysis of Part B of this section assumed that the system could be considered LTI over a short time period T. If the time period T is short enough, a non-linear system may be approximated by a linear system. To do so only requires that the non-linear elements in A and B be approximated by versions of those elements that are linearized about the instantaneous operating point. This is where the "adaptive" part of the controller comes in. Sensors continuously measure the system state variables. From the state variables, we may infer CPL power using the system differential equations, Ohm's law and the relationship between power, voltage and current. Given the measured CPL power, the CPL impedance is periodically linearized. Using the linearized impedance of the CPL in

place of the actual non-linear CPL impedance permits us to form the linear differential equations needed to perform LQR. As the CPL impedances are periodically linearized, the LQR controller optimal feedback gains and input values adapt to changing CPL impedance.

2.   Execution Cycle

To better illustrate LQR-SM, we describe the full execution cycle for one sub-cycle. The execution cycles described below are shown in Figure 13, and the controller's block diagram is shown in Figure 14.
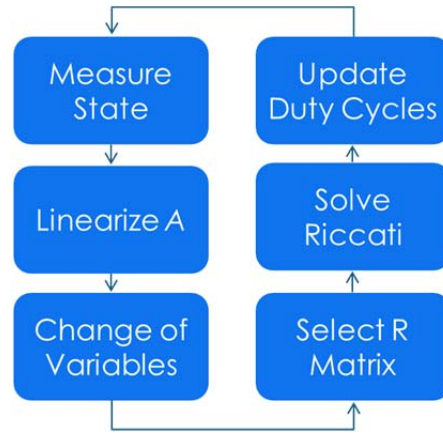
Figure 13. LQR-SM Execution Cycle
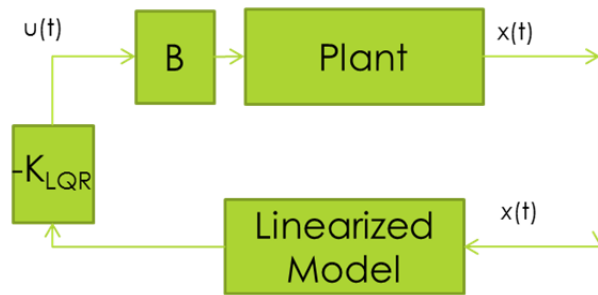
Figure 14. Control Block Diagram

The first step in the process is to derive the state-space model of the plant. Starting with the physical model of the plant, we derive the system of first-order

differential equations. If the system includes non-linear differential equations, these equations must be linearized about the instantaneous operating point. The system must also be fully observable; this requirement facilitates both the use of LQR as well as the calculations required to perform necessary linearization of the differential equations.

Once the state-space model of the system is derived, we implement a change of state-variables. LQR regulates all state-variables to the zero condition; thus, any state variables with a non-zero steady-state value need to be level-shifted to zero.

Take, for example, the state variable I_L, which is the current through an inductor, for an electrical system. The current I_L should be decomposed into its DC and small-signal values: $I_L = i_{L\text{-}ss} + I_{L\text{-}DC}$. The state variable that we want to regulate to zero is $i_{L\text{-}ss}$; therefore, we perform a change of state variables to define our regulated state variable as $i_{L\text{-}ss} = I_L - I_{L\text{-}DC}$. We perform this change of variables for all of our state variables, making appropriate changes to the A and B matrices. This allows the designer to essentially deal with the small-signal and DC portions of the control problem separately. Now, we have the A and B matrices as well as the state-vector x for the sub-cycle of interest.

Before we can solve the Riccati equation, Equation (4.6), and obtain our input via Equation (4.7), we must select the appropriate R matrix for the switching cycle. There may be up to N R-matrices for a multi-rate system with N sub-cycles; however, in practice there are often fewer. Take, for example, a system where one input is updated every sub-cycle while a second input value is updated only every four sub-cycles. With four sub-cycles per super-cycle, we could maintain $R_1, R_2, \ldots, R_N$ matrices, but this is not necessary since R_2–R_4 are in fact identical. For a system implemented on a microcontroller, this represents a memory savings over LQR-PD. This advantage in speed and memory extends even further when we consider systems that have large numbers of sub-cycles per super-cycle. LQR-PD computational complexity grows as the square of the number of sub-cycles. We see that N sub-cycles results in an $N{\times}N$ matrix of matrices for LQR-PD, while the computational complexity of LQR-SM remains constant. The only additional cost is in the memory required to store additional R matrices and the software to sequence those matrices appropriately.

We assume a constant *Q* matrix that is common to all sub-cycles. The selection of appropriate *Q* and *R* matrices is left for a later section.

Now that *A*, *B*, *Q*, *R*, and *x* are determined for the sub-cycle, we input these values into Equation (4.6) and obtain the small-signal input values *u* from Equation (4.7). If any of our input values also have a DC component, the DC value must be added to the small-signal value obtained from Equation (4.7) in order to obtain the total desired input value.

Now that our input values are known, we can update the device input values. Switching type devices, such as DC-DC converters, are typically governed by a pulse-width modulated (PWM) duty cycle. For such a case, we must convert the desired device output value to the appropriate PWM duty cycle for that device. Any devices that are not updated during the computation cycle are assumed to hold a constant output value or constant duty cycle.

3. Rotating B-Matrix

Previously, we described a method of LQR-SM where a common *B* matrix is used and the *R*-matrix is selected for the specific sub-cycle combination of adjustable inputs. An alternate method of LQR-SM is alike in every way except *B*-matrices are developed for each unique combination of adjustable inputs and the *R*-matrix is held common among all subcycles. In this implementation, the *B*-matrices developed are identical to those developed for LQR-PD where the non-adjustable, restricted, inputs during the sub-cycle have their *B*-values set to zero.

The rotating *B*-matrix implementation of LQR-SM produces results much closer to those produced by LQR-PD but suffers from reduced ROAs. The exact mechanism for this reduction in ROA is not presently known. We conjecture that rotating *B* LQR-SM has a more rigid structure owing to the additional zeroes in the *B* matrices. Unlike the rotating *R* implementation of LQR-SM, the Riccati solver has less trade-space computing the optimal state trajectory. This increased rigidity may result in greater "brittleness," which is manifest as smaller ROAs.

D. COMPARISON EXAMPLE

Now that we have a basic description of LQR-SM, we next desire to compare its performance to LQR-PD and determine the strengths and weaknesses of each controller relative to one another. To do so, we construct an example problem and compare the results. We then use these results to justify the conclusion that LQR-SM's differences in performance are outweighed by its advantages in computational complexity. The comparison performed in this section was originally presented in [39].

1. Test Model

The example we use to illustrate the differences between LQR-SM and LQR-PD is of a truncated version of the NAVSEA circuit model developed in Chapter III. Figure 15 is an illustration of the truncated circuit model. The truncated system is composed of two PGMs. One PGM is rated to deliver 40.0 MW of power, while the second PGM is rated to deliver 10.0 MW of power. PGMs are modeled as controlled voltage sources with an RL output impedance. PGMs are connected directly to a 12.0-kV MVDC bus. The PGMs each operate at a switching speed of 1.0 kHz; thus, the PGM inputs are adjusted every 1.0 ms.

The bus has a buffer capacitor. Two load zones are connected to the MVDC bus. Each load zone is composed of input cabling modeled as an RLC section, an RC input filter, an average-value model of a buck converter, a CPL, and a HESS. The CPL is modeled as a controlled current source whose current is equal to load power divided by the zone bus voltage. In this way, the power delivered to the CPL is constant. The HESS is modeled as a controlled current source. Both HESS are switched at 8.0 kHz, resulting in an update period 0.125 ms.

The first load zone is regulated to maintain 1.0 kV with a CPL maximum loading of 20.0 MW. The second load zone is regulated to maintain 6.0 kV with a CPL maximum loading of 28.0 MW. The system is initially at steady-state with Zone #1 loaded at 15.0 MW and Zone #2 loaded at 9.0 MW. This represents the 50% load condition. At 0.0 seconds, the loading is instantaneously stepped to 20.0 MW in Zone #1 and 28.0 MW

in Zone #2. This represents the 100% load condition. At 20.0 ms, the loading is instantaneously stepped back to the 50% load condition. The Q and R matrices are tuned to ensure that the voltage transients do not exceed ±10% of their steady-state values. Circuit component values are displayed in Table 1.

Table 1.   Example Circuit Component Values

R_{g1}	0.25 Ω	L_{z1}	70.5 μH	C_{d1}	1.7 μF
R_{g2}	0.30 Ω	L_{z2}	47.0 μH	C_{d2}	2.3 μF
L_{g1}	2.00 mH	C_{z1}	2.46 μF	L_{b1}	30.6 μH
L_{g2}	1.80 mH	C_{z2}	3.69 μF	L_{b2}	926 μH
C_{bus}	4.0 μF	R_{d1}	10 Ω	C_{b1}	75 mF
R_{z1}	3.30 mΩ	R_{d2}	10 Ω	C_{b2}	1.25 mF
R_{z2}	2.20 mΩ				

Figure 15. Two-PGM, Two-Zone Experimental Circuit Model

2.   Controller Implementation

The first step in developing the controller is to derive the differential equations for the circuit in Figure 15. These differential equations are given as

$$dI_{g1} = \left(\frac{1}{L_{g1}}\right)\left(E_1 - R_{g1}I_{g1} - V_{bus}\right)$$

$$dI_{g2} = \left(\frac{1}{L_{g2}}\right)\left(E_2 - R_{g2}I_{g2} - V_{bus}\right)$$

$$dV_{bus} = \left(\frac{1}{C_{bus}}\right)\left(I_{g1} + I_{g2} - I_{z1} - I_{z2}\right)$$

$$dI_{z1} = \left(\frac{1}{L_{z1}}\right)\left(V_{bus} - R_{z1}I_{z1} - V_{z1}\right)$$

$$dI_{z2} = \left(\frac{1}{L_{z2}}\right)\left(V_{bus} - R_{z2}I_{z2} - V_{z2}\right)$$

$$dV_{z1} = \left(\frac{1}{C_{z1}}\right)\left(I_{z1} - \frac{V_{z1} - V_{d1}}{R_{d1}} - D_1 I_{b1}\right)$$

$$dV_{z2} = \left(\frac{1}{C_{z2}}\right)\left(I_{z2} - \frac{V_{z2} - V_{d2}}{R_{d2}} - D_2 I_{b2}\right)$$

$$dV_{d1} = \left(\frac{1}{C_{d1}}\right)\left(\frac{V_{z1} - V_{d1}}{R_{d1}}\right)$$

$$dV_{d2} = \left(\frac{1}{C_{d2}}\right)\left(\frac{V_{z2} - V_{d2}}{R_{d2}}\right)$$

$$dI_{b1} = \left(\frac{1}{L_{b1}}\right)\left(D_1 V_{z1} - V_{b1}\right)$$

$$dI_{b2} = \left(\frac{1}{L_{b2}}\right)\left(D_2 V_{z2} - V_{b2}\right)$$

$$dV_{b1} = \left(\frac{1}{C_{b1}}\right)\left(I_{b1} - \frac{P_1}{V_{b1}} + I_{s1}\right)$$

$$dV_{b2} = \left(\frac{1}{C_{b2}}\right)\left(I_{b2} - \frac{P_2}{V_{b2}} + I_{s2}\right)$$

$$, \qquad (4.15)$$

where E_x is the PGM voltage for machine "x"; I_{gx} is the series inductor current for PGM "x"; V_{bus} is the MVDC bus voltage; I_{zx} is the line current to zone "x"; V_{zx} is the voltage at the input to buck converter for zone "x"; V_{dx} is the voltage across the damper capacitor for zone "x"; I_{bx} is the buck inductor current for zone "x"; V_{bx} is the voltage on the buck filter capacitor for zone "x"; I_{sx} is the current injected from HESS "x"; P_x is the CPL power in zone "x," and D_x is the duty cycle for the buck converter for zone "x."

After linearization about the instantaneous operating point, the small-signal value of CPL resistance is represented as the resistance

$$R_{CPL} = -\frac{V_{CPL}^2}{P_{CPL}}.$$

$$(4.16)$$

The state-space representation of the small-signal system becomes the linear equation

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{4.17}$$

where

$$A = \begin{bmatrix}
\dfrac{-R_{g1}}{L_{g1}} & 0 & \dfrac{1}{L_{g1}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \dfrac{-R_{g2}}{L_{g2}} & \dfrac{1}{L_{g2}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\dfrac{1}{C_{bus}} & \dfrac{1}{C_{bus}} & 0 & \dfrac{-1}{C_{bus}} & \dfrac{-1}{C_{bus}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \dfrac{1}{L_{z1}} & \dfrac{-R_{z1}}{L_{z1}} & 0 & \dfrac{-1}{L_{z1}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \dfrac{1}{L_{z2}} & 0 & \dfrac{-R_{z2}}{L_{z2}} & 0 & \dfrac{-1}{L_{z2}} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \dfrac{1}{C_{z1}} & 0 & \dfrac{-1}{C_{z1}R_{d1}} & 0 & \dfrac{1}{C_{z1}R_{d1}} & 0 & \dfrac{-D_1}{C_{z1}} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \dfrac{1}{C_{z2}} & 0 & \dfrac{-1}{C_{z2}R_{d2}} & 0 & \dfrac{1}{C_{z2}R_{d2}} & 0 & \dfrac{-D_2}{C_{z2}} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \dfrac{1}{C_{d1}R_{d1}} & 0 & \dfrac{-1}{C_{d1}R_{d1}} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \dfrac{1}{C_{d2}R_{d2}} & 0 & \dfrac{-1}{C_{d2}R_{d2}} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \dfrac{D_1}{L_{b1}} & 0 & 0 & 0 & 0 & 0 & \dfrac{-1}{L_{b1}} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \dfrac{D_2}{L_{b2}} & 0 & 0 & 0 & 0 & 0 & \dfrac{-1}{L_{b2}} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dfrac{1}{C_{b1}} & 0 & \dfrac{-1}{C_{b1}R_{cpl1}} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dfrac{1}{C_{b2}} & 0 & \dfrac{-1}{C_{b2}R_{cpl2}}
\end{bmatrix}$$

and

$$B = \begin{bmatrix}
\dfrac{1}{L_{g1}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \dfrac{1}{L_{g2}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dfrac{1}{C_{b1}} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dfrac{1}{C_{b2}}
\end{bmatrix}.$$

45

a. LQR-PD

Referring to Section B of this chapter, to implement LQR-PD we need to take the sub-cycle matrices and form super-cycle, block-cyclic matrices.

The first step in this process is to linearize the differential equations to form the linear continuous-time matrices A and B. Next, we form the discrete-time matrices F and G from the continuous-time matrices A and B. We do this by recognizing the relationship between the discrete-time difference equation and the continuous-time differential equation. We convert the continuous-time differential equation of Equation (4.1) into the discrete-time representation of Equation (4.8) using the time-step between switching events Δt by the relationships of

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$F = I + A \cdot \Delta t$$
$$G = B \cdot \Delta t$$
$$x[k+1] = Fx[k] + Gu[k] \tag{4.18}$$

We now have the F and G matrices we need to form the block-cyclic $\hat{A}$ and $\hat{B}$ matrices. For the system described by Figure 15, the voltage sources, E_1 and E_2, are switched at a rate of 1.0 kHz on alternating half-cycles. The current sources, I_{s1} and I_{s2}, are switched at a rate of 8.0 kHz; thus, we can decompose the control into eight 0.125-ms sub-cycles per each 1.0-ms super-cycle. During the first sub-cycle, we switch E_1, I_{s1} and I_{s2}. During the fifth sub-cycle, we switch E_2, I_{s1}, and I_{s2}. For sub-cycles two, three, four, six, seven, and eight, we switch only I_{s1} and I_{s2}. Switching in this manner forms three distinct G matrices: G_A is defined for the first sub-cycle, G_B is defined for the fifth sub-cycle, and G_C is defined for all other sub-cycles. The matrix G_A has the same structure as the B matrix described in Equation (4.17), but $G_{A2,2}$ is zero. In a similar way, G_B is also analogous to B, but $G_{B1,1}$ is zero, and G_C is also like B, but $G_{C1,1}$ and $G_{C2,2}$ are zero.

With F and the set of G matrices known, we may form the block-cyclic state and feedback matrices $\hat{A}$ and $\hat{B}$ according to the process described in Section B of this chapter. The $\hat{Q}$ and $\hat{R}$ matrices are formed with each entry given by

$$Q=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2^{10.2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2^{5.4} \end{bmatrix} R=\begin{bmatrix} 15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & .1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & .1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & .1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .5 \end{bmatrix}. \quad (4.19)$$

Once $\hat{A}, \hat{B}, \hat{Q}$ and $\hat{R}$ are formed, Equation (4.11) can be solved with MATLAB *dare()* and gain matrices calculated with Equation (4.13). Once the gain matrices are known, then the state vector *x[k]* can be multiplied by the appropriate gain matrix for the sub-cycle to find the desire input values.

Now that the small-signal input values have been found, we add them to the steady-state values for the inputs in order to determine the total commanded voltage or current for the input devices. To enforce load sharing between the two voltage sources, we set their values as given by

$$E_1[k]=V_{ref}+0.80\frac{P_{CPL1}+P_{CPL2}}{V_{ref}}R_{g1}+u_1[k]$$

$$E_2[k]=V_{ref}+0.20\frac{P_{CPL1}+P_{CPL2}}{V_{ref}}R_{g2}+u_2[k].$$

$$I_{s1}[k]=u_{12}[k]$$

$$I_{s2}[k]=u_{13}[k]$$

(4.20)

This ensures that PGM #1 and PGM #2 share load according to their ratings and that MVDC bus voltage is maintained at V_{ref} for steady-state operation. PGM #1 provides 80% of total load current at steady state, while PGM #2 provides the remaining 20%. The HESS current sources have DC values of zero; therefore, the total current demanded is simply the small-signal current demand.

b. *LQR-SM Rotating R*

To implement LQR-SM control by using rotating R-matrices, we can re-use many of the elements found in LQR-PD. The A and B matrices of Equation (4.17) remain the same, as does the Q matrix of Equation (4.19). Where LQR-PD defined G_A, G_B and G_C, we similarly define R_A, R_B and R_C; R_A is defined for the first sub-cycle, R_B is defined for the fifth sub-cycle, and R_C is defined for all other sub-cycles. The R_A matrix has the same structure as the R matrix described in Equation (4.19), but $R_{A2,2}$ is set to 10^3. In a similar way, R_B has the form of R, but $R_{B1,1}$ is set to 10^3. Likewise, R_B is has the form of R, but $R_{C1,1}$ and $R_{C2,2}$ are set to 10^3. The full sequence of R-matrices in the super-cycle is displayed in Table 2.

Unlike LQR-PD, which is only linearized every super-cycle, LQR-SM is linearized every sub-cycle. After linearization about the instantaneous operating point, for a given sub-cycle we know A, B, Q, and R. Equation (4.6) is solved for the sub-cycle with inputs calculated by Equations (4.7) and (4.20).

Table 2. LQR-SM Rotating R Super-Cycle

Super-cycle Sequence								
Sub-cycle	1	2	3	4	5	6	7	8
R matrix	R_A	R_C	R_C	R_C	R_B	R_C	R_C	R_C

c. *LQR-SM Rotating B*

To implement LQR-SM control by using rotating B matrices, we define B_A, B_B, and B_C in much the same way as we defined G_A, G_B, and G_C for LQR-PD. The B_A matrix has the same structure as the B matrix described in Equation (4.17), but $B_{A2,2}$ is zero. In a similar way, B_B has the form of B, but $B_{B1,1}$ is zero. Likewise, B_C has the form of B, but $B_{C1,1}$ and $B_{C2,2}$ are zero. The full sequence of B-matrices in the super-cycle is displayed in Table 3. The A matrix is the same as in the rotating R implementation and is linearized every sub-cycle. The Q and R matrices are identical to those in Equation (4.19). As

before, with *A, Q, R* and the appropriate *B* for the sub-cycle selected, we solve Equation (4.6) and calculate the inputs for the sub-cycle by Equations (4.7) and (4.20).

Table 3.   LQR-SM Rotating *B* Super-Cycle

Super-cycle Sequence								
Sub-cycle	1	2	3	4	5	6	7	8
B matrix	B_A	B_C	B_C	B_C	B_B	B_C	B_C	B_C

3.   Results

Results comparing the performance of LQR-PD and both implementations of LQR-SM are described below.

a.   LQR-PD versus LQR-SM Rotating R-Matrix Implementation

In Figure 16a, PGM voltages are compared between the LQR-PD and LQR-SM controllers. The LQR-PD controller employs a wider range of PGM voltages with more oscillation than the LQR-SM controller. Despite this wider dynamic range of PGM voltage, the currents through the PGM output inductors, shown in Figure 16b, are very similar. These greater voltage oscillations manifest in greater variation in MVDC bus voltage and a longer MVDC voltage settling time for the LQR-PD controller, as illustrated in Figure 17.
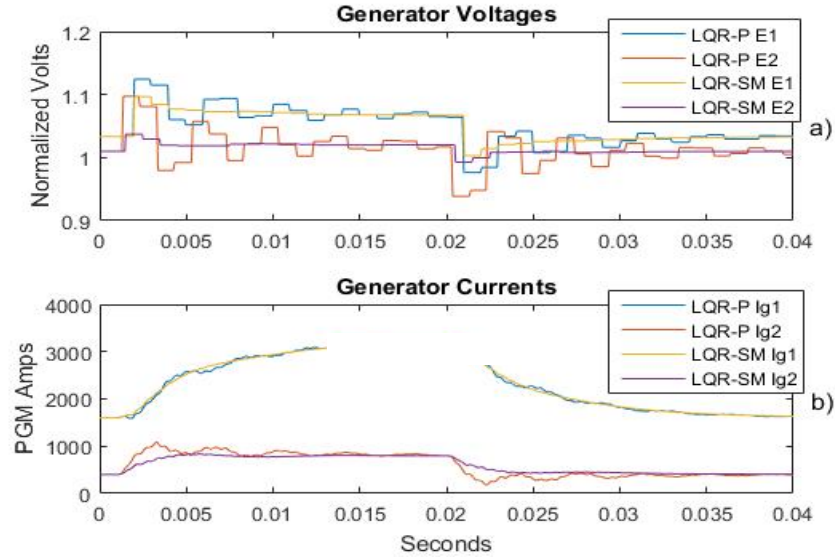
Figure 16. PGM Normalized Voltages (a) and Currents (b)
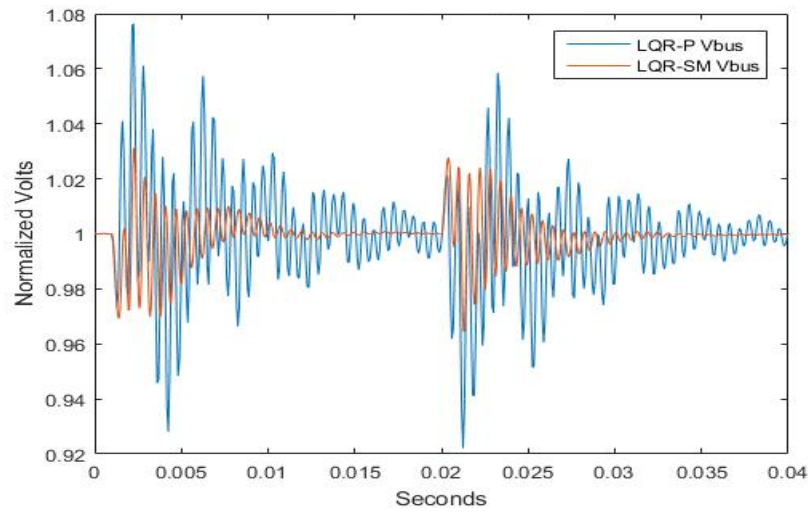
Figure 17. MVDC Bus Voltage

When we compare the voltages in each of the load zones, shown in Figure 18a, we see that the voltages match very closely. Settling times have no discernible difference. The buck inductor current transients in both zones, shown in Figure 18b, also match quite closely. The LQR-PD current transient has a high-frequency oscillation. This oscillation

may be a response by the HESS#2 current to counter the larger MVDC bus oscillations produced by the LQR-PD controller.
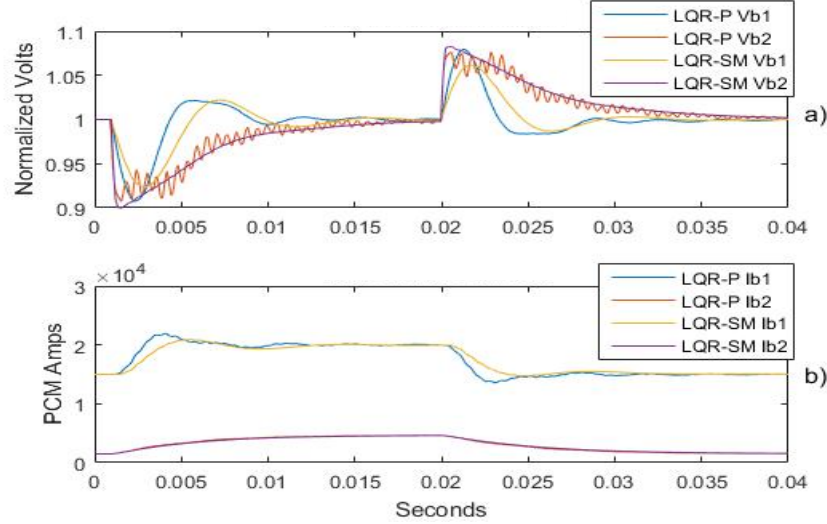
Figure 18. Normalized Zone Buck Voltages (a) and Currents (b)

The oscillations exerted by the LQR-PD controller are also reflected in the HESS #2 current transient of Figure 19a. Despite what appears to be greater control effort, the LQR-PD controller requires 60% less HESS energy in Zone #1 and 8% less HESS energy in Zone #2 to stabilize the transient. The HESS energy levels are displayed in Figure 19b.

An analysis of regions of attraction yields more comparisons. For each of the ROAs shown in Figures 20 and 21, the circuit of Figure 15 is at steady-state, operating at 100% loading in both zones. For Figure 20, the Zone #1 current and voltage were disturbed in the region shown. The circuit and controller responses were evaluated. If the circuit returned to steady-state operation without allowing bus voltage to dip below 0.0 V, without experiencing a Riccati solution error, or without exceeding a 40.0-ms settling time, the disturbance was evaluated as within the ROA. Both Figures 20 and 21 show no significant difference in ROA between the two controllers.
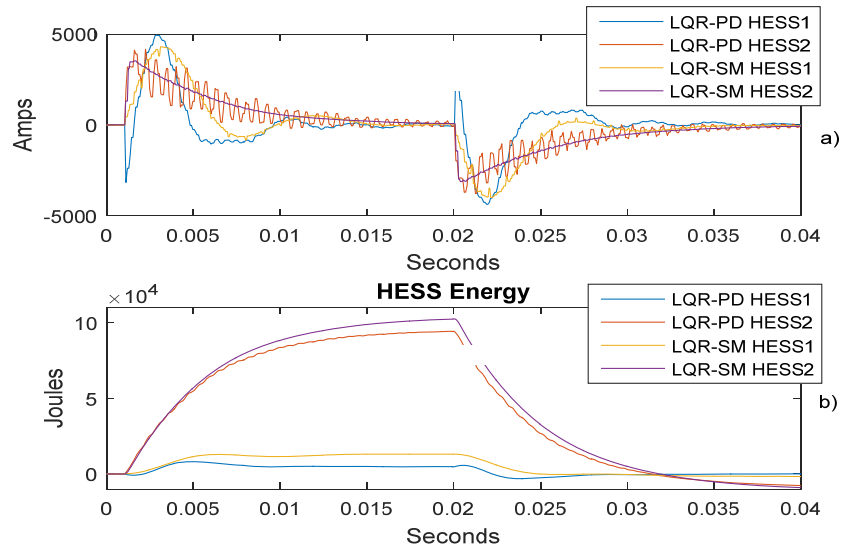
Figure 19. HESS Current (a) and Delivered Energy (b)
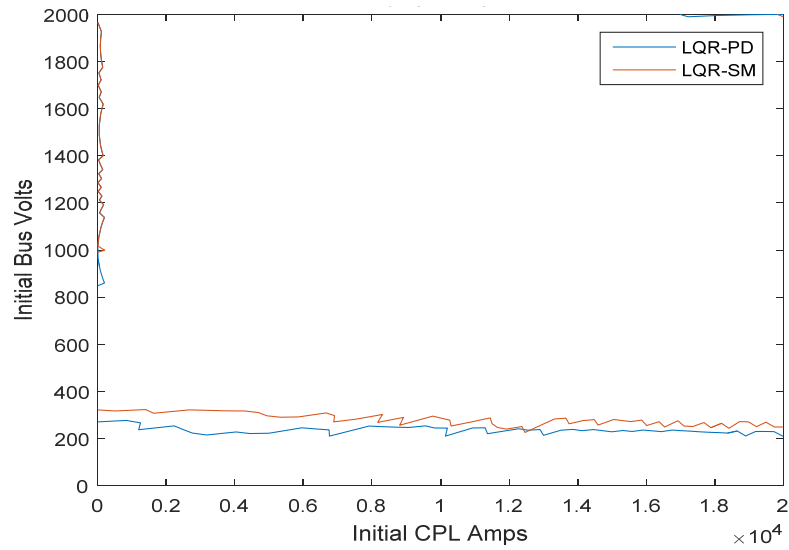
Figure 20. Zone #1 Region of Attraction
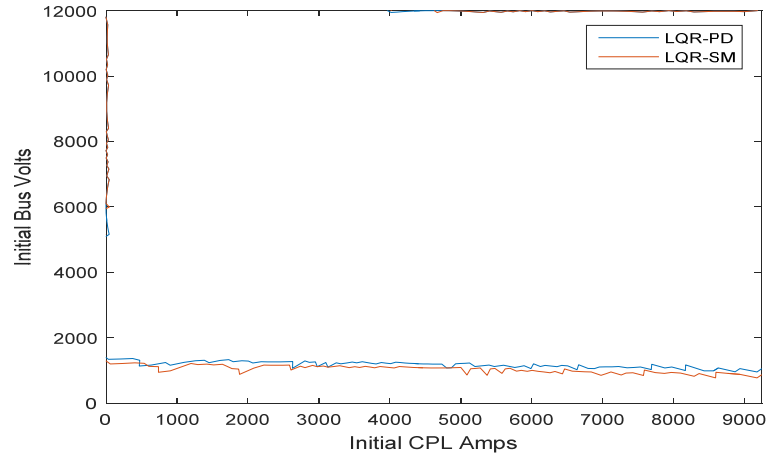
Figure 21. Zone #2 Region of Attraction

b. LQR-PD versus LQR-SM Rotating B-Matrix Implementation

When we examine the simulation results for LQR-SM with Rotating *B* matrices versus LQR-PD, we see that the two controllers behave nearly identically to one another. The PGM voltages, presented in Figure 22a, commanded by each controller are very similar while the PGM output inductor currents, presented in Figure 22b, are nearly identical.
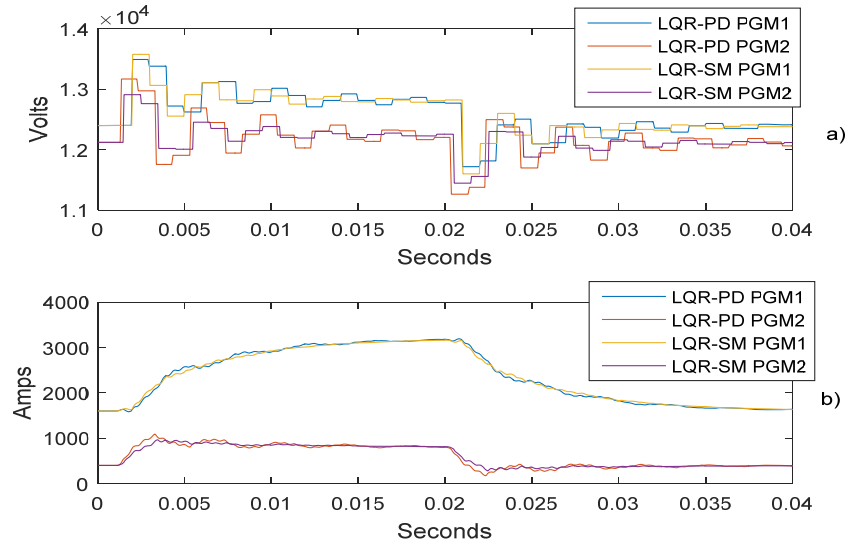
Figure 22. PGM Voltages (a) and Currents (b)

This similarity of PGM voltages and currents extends further when we examine the MVDC bus transients as shown in Figure 23. Here again, the differences between the two controllers are minor. The LQR-SM controller has a slightly smaller range of MVDC bus voltages and a quicker settling time. The zone voltages and buck inductor currents, presented in Figure 24, are nearly identical for both controllers. Inspection of the HESS currents and delivered energies in Figure 25 shows that these two controllers are, again, very similar with LQR-PD; both controllers stabilize the transient with marginally more HESS energy.
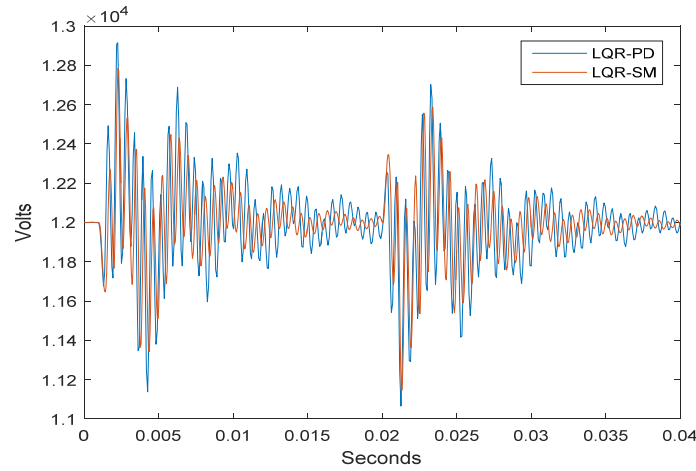
Figure 23. MVDC Bus Voltage
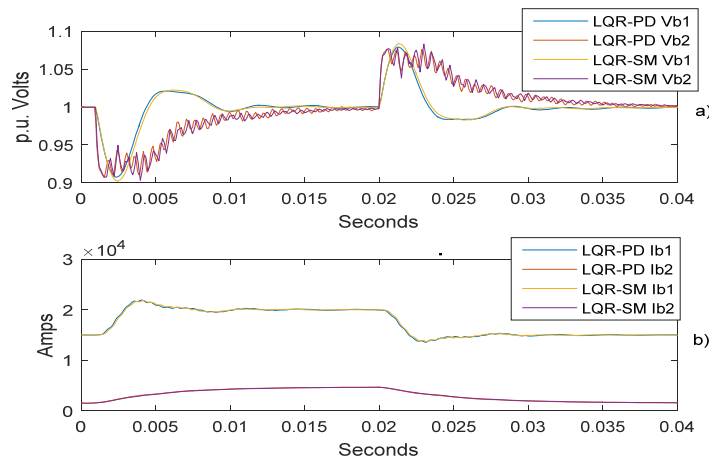
Figure 24. Normalized Zone Buck Voltages (a) and Currents (b)
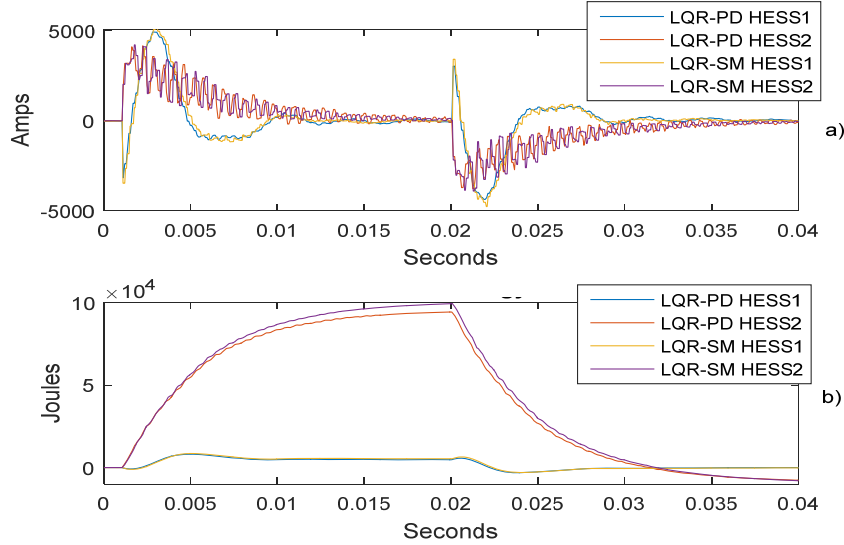
Figure 25. HESS Currents (a) and Delivered Energy (b)

The major difference between the two controllers appears when we examine the ROAs presented in Figures 26 and 27. Both Figures 26 and 27 we see that, in the investigated region, the ROA for LQR-SM with Rotating B matrices is smaller than for LQR-PD. Comparing the ROAs in Figures 20 and 21 to Figures 25 and 26, respectively, we notice that the LQR-SM with Rotating B matrices has smaller ROAs than both LQR-PD and LQR-SM with rotating R matrices.

The regions that are unstable for LQR-SM with Rotating B but stable for LQR-SM with Rotating R fail the 40.0-ms settling time test. As we have seen in Figures 23–25, the Rotating B control takes a bit longer to settle than the Rotating R controller, which is the most likely reason why extreme disturbances fail the 40.0-ms settling time test. Allowing a longer settling time may demonstrate that these regions can converge to steady-state values.
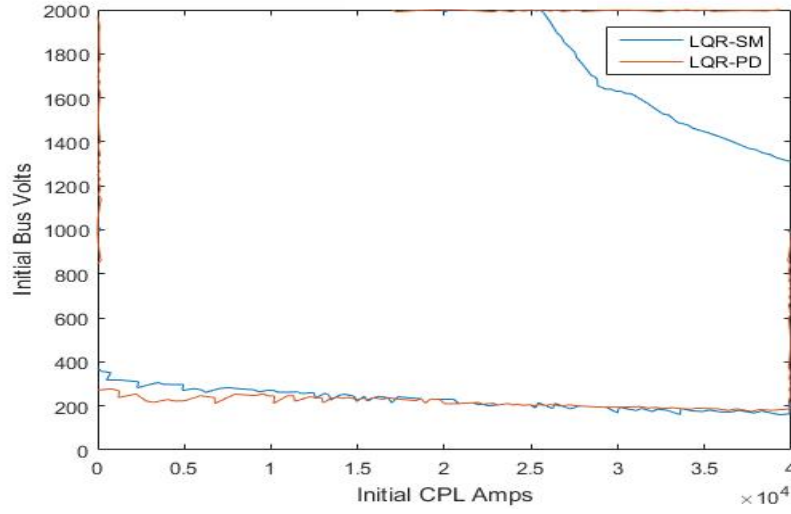
Figure 26. Zone #1 Region of Attraction
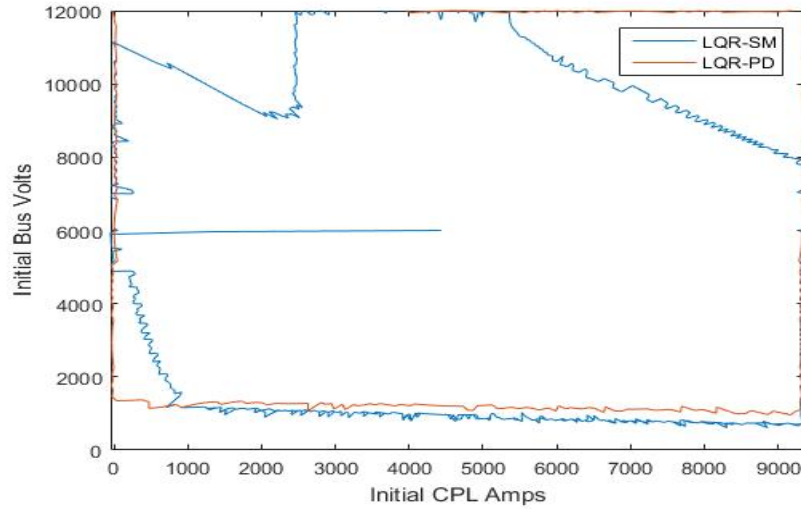
Figure 27. Zone #2 Region of Attraction

As mentioned before, there is a difference in the computational complexity of the competing controllers. LQR-PD requires solving the Riccati equation for very large matrices. For a controller applied to an $N{\times}N$ state-space system with L sub-cycles, the Riccati solver must resolve a solution for $NL{\times}NL$ matrices once every L sub-cycles. The LQR-SM approach only requires the Riccati solver to resolve a solution for $N{\times}N$ matrices but does so every sub-cycle. For the simulation trials presented in this section, MATLAB

is able to complete the LQR-PD simulation in about 27 seconds. For exactly the same system, the LQR-SM method can produce simulation results in just 1.9 seconds.

Attempts to expand this analysis to a larger model of a twentieth-order circuit problem, as shown in Figure 12, failed. LQR-PD could not be implemented at the twentieth-order level due to the MATLAB Riccati solver's inability to resolve a solution. At the thirteenth-order level, which is presented above, the MATLAB *dare()* function produced warnings regarding "ill-conditioned matrices" and "eigenvalues too close to the origin."

E.   SUMMARY

LQR-SM is a multi-input, multi-rate LQR-based control. LQR-SM can be implemented as either a rotating *R* matrix version or a rotating *B* matrix version. Both versions of LQR-SM controllers produce similar results to LQR-PD but do so much more quickly. Due to LQR-SM linearizing the state matrix more frequently, both LQR-SM controllers provide slightly better regulation of the MVDC bus than LQR-PD with faster settling times. The Rotating *R* version of LQR-SM has the best MVDC bus regulation and fastest settling times of the three controllers discussed in this section. LQR-PD used less HESS energy to stabilize the example transient than the LQR-SM controllers. Here, we find a trade-off between controllers. LQR-PD is computationally more complex but produces a more "optimal" solution by using less input energy to achieve the desired regulation. LQR-SM is computationally less complex but trades that simplicity for a less optimal use of input energy.

Region of attraction analysis showed that LQR-PD and LQR-SM with rotating *R* matrices had nearly identical ROAs but that the Rotating *B* version of LQR-SM had smaller ROAs.

The final advantage presented by LQR-SM over the classical multi-rate LQR of LQR-PD is robustness. Solving high-order Riccati equations often proves difficult, especially when the matrices input to the Riccati solver are as sparse as ours. By keeping

the matrices smaller, LQR-SM does not experience the computational problems encountered when using LQR-PD.

V. STOCHASTIC SEARCH DESIGN

In this chapter, we investigate two different stochastic search algorithms: simulated annealing and the genetic algorithm. Stochastic searches are useful tools for exploring high-order search spaces and non-linear systems. Here, stochastic search design is used to design an LQR-SM controller of the rotating R-matrix type that meets certain transient dynamic specifications while minimizing the stored energy requirement for the NAVSEA circuit model of Figure 12.

LQR controllers present a challenge due to their large number of design variables. Each entry in the Q and R matrices represents a free variable that must be assigned a value. The requirement that Q is positive semi-definite and R is positive-definite restricts choices to some degree but still leaves quite a bit of guesswork for the designer. A stochastic search algorithm, when properly constrained, can provide the necessary variable values. The automated nature of a stochastic search allows the designer to set the algorithm to work, turn his or her attention to other tasks, then return to check the results of the search. This is a much more efficient use of human resources than iterative design.

In this chapter, we first discuss the state-space model of the system under study. Next, we develop the performance criteria that are used to evaluate each of our candidate solutions. Finally, we implement both simulated annealing and genetic algorithm searches to design the optimum LQR-SM controller for the system.

A.   SYSTEM MODEL

The system model used is for the average-value model developed in Chapter III. For review, there are four power generation modules (PGM), two of which are 40.0 MW while the other two are 10.0 MW for a total of 100.0 MW of generating capacity. The main MVDC bus is regulated to 12.0 kV. There are three load zones connected to the 12.0-kV MVDC bus. Each load zone consists of a bus impedance derived from [27], a series-damped RC filter in parallel with the medium voltage side of a DC-DC buck converter, the buck converter, an ideal CPL, and an ideal controlled current-source HESS. The CPL and HESS are parallel connected to the low-voltage side of the buck

converter. The three load zones are a 20.0-MW capacity CPL on a 1.0-kV bus, 30.0-MW capacity CPL on a 6.0-kV bus, and 80.0-MW capacity CPL on a 10.0-kV bus. The first two zones have HESS, but the third zone does not. Total load capacity exceeds the total generating capacity; however, overload conditions are not in the scope of this study. The circuit model was developed in Chapter III.B and first illustrated in Figure 12; here, we repeat the illustration as Figure 28 for the reader's convenience.
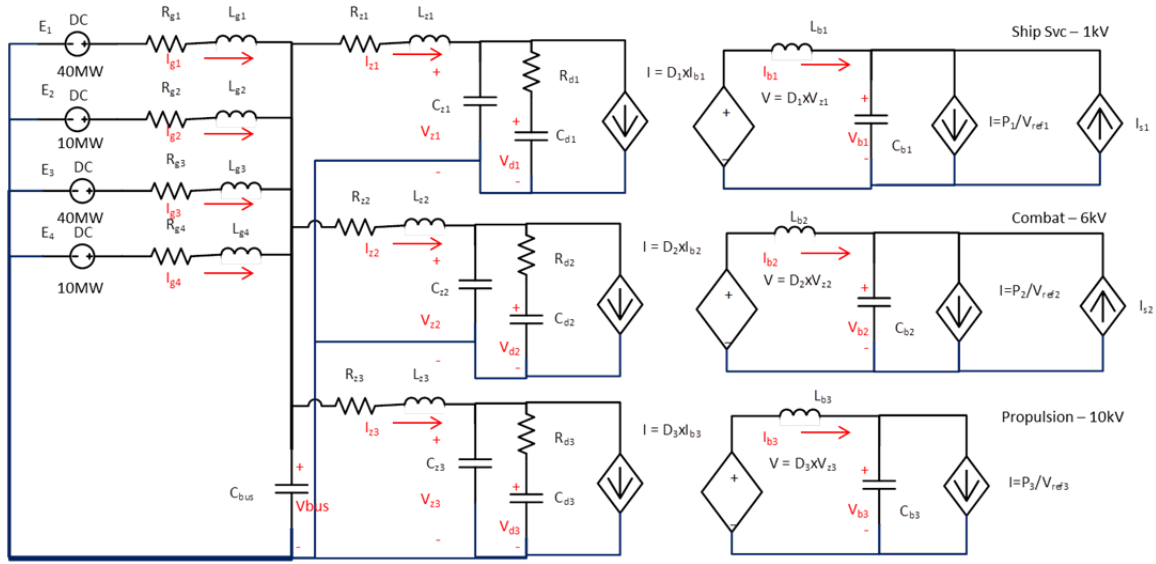
Figure 28. Distribution System Circuit Model, Repeated from Figure 12

The circuit model of Figure 28 yields the differential equations

$$dI_{g1} = \left(\frac{1}{L_{g1}}\right)\left(E_1 - R_{g1}I_{g1} - V_{bus}\right)$$

$$dI_{g2} = \left(\frac{1}{L_{g2}}\right)\left(E_2 - R_{g2}I_{g2} - V_{bus}\right)$$

$$dI_{g3} = \left(\frac{1}{L_{g3}}\right)\left(E_3 - R_{g3}I_{g3} - V_{bus}\right) \tag{5.1a}$$

$$dI_{g4} = \left(\frac{1}{L_{g4}}\right)\left(E_4 - R_{g4}I_{g4} - V_{bus}\right)$$

$$dV_{bus} = \left(\frac{1}{C_{bus}}\right)\left(I_{g1} + I_{g2} + I_{g3} + I_{g4} - I_{z1} - I_{z2} - I_{z3}\right)$$

$$dI_{z1} = \left(\frac{1}{L_{z1}}\right)\left(V_{bus} - R_{z1}I_{z1} - V_{z1}\right)$$

$$dI_{z2} = \left(\frac{1}{L_{z2}}\right)\left(V_{bus} - R_{z2}I_{z2} - V_{z2}\right)$$

$$dI_{z3} = \left(\frac{1}{L_{z3}}\right)\left(V_{bus} - R_{z3}I_{z3} - V_{z3}\right)$$

$$dV_{z1} = \left(\frac{1}{C_{z1}}\right)\left(I_{z1} - I_{d1} - D_1I_{b1}\right)$$

$$dV_{z2} = \left(\frac{1}{C_{z2}}\right)\left(I_{z2} - I_{d2} - D_2I_{b2}\right)$$

$$dV_{z3} = \left(\frac{1}{C_{z3}}\right)\left(I_{z3} - I_{d3} - D_3I_{b3}\right)$$

$$dV_{d1} = \left(\frac{1}{C_{d1}}\right)\left(\frac{V_{z1} - V_{d1}}{R_{d1}}\right)$$

$$dV_{d2} = \left(\frac{1}{C_{d2}}\right)\left(\frac{V_{z2} - V_{d2}}{R_{d2}}\right)$$

$$dV_{d3} = \left(\frac{1}{C_{d3}}\right)\left(\frac{V_{z3} - V_{d3}}{R_{d3}}\right)$$

$$dI_{b1} = \left(\frac{1}{L_{b1}}\right)\left(D_1V_{z1} - V_{b1}\right)$$

$$dI_{b2} = \left(\frac{1}{L_{b2}}\right)\left(D_2V_{z2} - V_{b2}\right)$$

$$dI_{b3} = \left(\frac{1}{L_{b3}}\right)\left(D_3V_{z3} - V_{b3}\right)$$

$$dV_{b1} = \left(\frac{1}{C_{b1}}\right)\left(I_{b1} - \frac{P_1}{V_{b1}} + I_{s1}\right)$$

$$dV_{b2} = \left(\frac{1}{C_{b2}}\right)\left(I_{b2} - \frac{P_2}{V_{b2}} + I_{s2}\right)$$

$$dV_{b3} = \left(\frac{1}{C_{b3}}\right)\left(I_{b3} - \frac{P_3}{V_{b3}}\right)$$

$$\text{, } \qquad (5.1b)$$

where E_x is the PGM voltage, I_{gx} is the PGM inductor current, V_{bus} is the MVDC bus voltage, I_{zx} is the line current to zone "x," V_{zx} is the voltage at the input to buck converter "x," V_{dx} is the damper capacitor voltage for zone "x," I_{bx} is the buck inductor current for zone "x" V_{bx} is the buck filter capacitor voltage for zone "x," and D_x is the duty cycle for the buck converter for zone "x." The necessary state variables for the LQR controller to properly regulate the system are given by

$$X_1 = I_{g1} - 0.40 I_0$$
$$X_2 = I_{g2} - 0.10 I_0$$
$$X_3 = I_{g3} - 0.40 I_0$$
$$X_4 = I_{g4} - 0.10 I_0$$
$$X_5 = V_{bus} - V_{ref}$$
$$X_6 = I_{z1} - I_{o1}$$
$$X_7 = I_{z2} - I_{o2}$$
$$X_8 = I_{z3} - I_{o3}$$
$$X_9 = V_{z1} - V_{ref} - R_{z1} I_{o1}$$
$$X_{10} = V_{z2} - V_{ref} - R_{z2} I_{o2}$$
$$X_{11} = V_{z3} - V_{ref} - R_{z3} I_{o3}$$
$$X_{12} = V_{d1} - V_{ref} - R_{z1} I_{o1}$$
$$X_{13} = V_{d2} - V_{ref} - R_{z2} I_{o2}$$
$$X_{14} = V_{d3} - V_{ref} - R_{z3} I_{o3}$$
$$X_{15} = I_{b1} - I_{ob1}$$
$$X_{16} = I_{b2} - I_{ob2}$$
$$X_{17} = I_{b3} - I_{ob3}$$
$$X_{18} = V_{b1} - V_{ref1}$$
$$X_{19} = V_{b2} - V_{ref2}$$
$$X_{20} = V_{b3} - V_{ref3}$$

$$, \tag{5.2}$$

where I_0 is the steady-state total MVDC current, V_{ref} is the MVDC bus reference voltage (12 kV), I_{o1} is the Zone #1 steady-state MVDC current (P_{CPL1}/V_{ref}), I_{o2} is the Zone #2 steady state MVDC current (P_{CPL2}/V_{ref}), I_{o3} is the Zone #3 steady-state MVDC current (P_{CPL3}/V_{ref}), I_{ob1} is the Zone #1 steady-state LVDC current (P_{CPL1}/V_{ref1}), I_{ob2} is the Zone #2 steady-state LVDC current (P_{CPL2}/V_{ref2}), I_{ob3} is the Zone #3 steady-state LVDC current (P_{CPL3}/V_{ref3}). The voltage V_{ref1} is the Zone #1 LVDC reference voltage (1.0 kV), V_{ref2} is the Zone #2 LVDC reference voltage (6.0 kV), and V_{ref3} is the Zone #3 LVDC reference voltage (10.0 kV).

B. EVALUATION CRITERIA

Next, we develop the evaluation criteria that are the basis for choosing the desirability, or fitness, of each candidate configuration. Each candidate configuration of *R*, *Q* and capacitor values is evaluated based on dynamic response to a presumed worst-case instantaneous load step from 50% total load power to 100% total load power. The power transient involves power in Zone #1 stepping from 15.0 MW to 20.0 MW, Zone #2 stepping from 5.0 MW to 30.0 MW, and Zone #3 stepping from 28.0 MW to 46.0 MW.

The dynamic performance specification selected is that MVDC bus voltage (V_{bus}) and the three zone buck voltages (V_{b1}, V_{b2}, and V_{b3}) shall remain within 10% of steady-state value during the transient. Given this dynamic performance specification, it is more convenient to graph the voltage transients with respect to the normalized voltage values (V_{bus}/V_{ref}, V_{b1}/V_{ref1}, etc.) so that the graph can be quickly analyzed to determine if the transient exceeds limits. A convenient benefit is that all of the voltage transients can be graphed on the same scale for ease of comparison and presentation.

Another specification placed on the transient is that harmonic content not exceed certain values. Since all PGMs and HESS interface through switch-type power electronics, we desire that the switching frequencies and certain harmonics not exceed -60 dB down in energy spectral density from the DC energy. This ensures relatively clean signal quality without excessive voltage ripple. We check for 1.0 kHz and 4.0 kHz harmonics from the PGMs that switch at 1.0 kHz each on staggered quarter cycles as well as 16.0 kHz from the HESS.

While the dynamic performance and harmonic content criteria discussed above are pass-or-fail criteria, our next criterion is a continuous scalar criterion: total system stored energy. The stored energy in the system is measured by the energy stored in capacitors in the steady-state condition plus the peak energy delivered by each HESS during the design transient. The full scoring function is described in the MATLAB function *fitness()* in Appendix A. The stored energy fitness score is given as

$$F_{itness} = \sum_{Cbus, Cd1, ..., Cb3} \frac{1}{2} C_x V_{C_x}^2 + \max(\int V_{b1} I_{HESS1} dt) + \max\left(\int V_{b2} I_{HESS2} dt\right) . \quad (5.3)$$

C.   SEED CANDIDATE DEVELOPMENT

Unlike the typical simulated annealing or genetic algorithm processes, we do not randomly choose the initial, or "seed," configuration of first-generation genomes. Instead, we must choose the seed configuration quite carefully. Because our cost function has both discrete and continuous components, it is not only possible but very likely that a randomly generated seed fails the discrete evaluation criteria. Subsequent trials, which are in the neighborhood of the unacceptable seed, are also very likely to be unacceptable. In this way, the stochastic search algorithm could potentially spend the entire set of iterations exploring an unacceptable region of the variable-space. To prevent this, we choose a seed that meets all of our dynamic and harmonic performance criteria. For our particular case, we choose the Q matrix and R matrix diagonal terms to be uniformly "1" in value. Capacitors are made large enough such that the dynamic performance criteria are met. By using large capacitors, we may simultaneously reduce overshoot and undershoot peaks, filter unwanted harmonics, and expand the sizes of ROAs.

In order to appropriately size the capacitors, we employ the theory for second-order systems with CPL discussed in the Chapter II. Detailed reasoning and calculations for developing the seed candidate are covered in Appendix B.

D.   SIMULATED ANNEALING

Simulated annealing is a stochastic search algorithm modeled after the metallurgical annealing process. In the annealing process, a hot piece of metal is repeatedly heated and cooled to allow the atoms within the alloy to form the crystalline structure with the lowest energy state. In theory, this minimum energy state has inter-atomic bonds with the least amount of built-in stress, therefore making the crystalline structure as strong as possible. At high temperature, atoms are free to bounce around and explore several energy states. As the system cools, the atoms become constrained to lower and lower energy states until temperature is so low that all atoms are frozen in place.

In simulated annealing, we define a function to measure the "energy" of our system or process. Along with this energy function, we also define a global non-physical "temperature" parameter. The simulated annealing process begins by randomly choosing the combination of variable values to form the initial energy state. Then, in successive iterations, the energy state is permitted to vary randomly through a neighborhood about the lowest observed energy state. As the "temperature" cools, the size of the neighborhood shrinks. The algorithm is then stopped when a certain energy level is reached, a certain temperature is reached, or a certain number of iterations have been performed [40].

For the implementation of simulated annealing used in this work, we start with a temperature of 100.0, have a period of constant temperature for 1600 iterations, slowly lower temperature over the next 800 iterations, then rapidly cool temperature over the final 800 iterations. Over the total sum of 3200 iterations, the algorithm is given a large number of iterations of high temperature in order to potentially escape a local minimum in search of the global minimum. Over the cooling period, the stochastic search is concentrated in a smaller and smaller neighborhood in order to find the absolute lowest energy state within the local area. A diagram of the simulated annealing process is featured in Figure 29. In Figure 29, we use λ^* to denote the lowest energy candidate, χ to denote the neighbor candidate, T for temperature, and J for the cost of the candidate. In this context, the terms "cost," "energy," and "score" are interchangeable.

In order to limit the search space, we define upper and lower bounds for each of our free variables. To generate the neighbor χ, we choose to "mutate" a portion of the design variables in λ^* by a factor with random normal distribution with a mean of one and variance equal to the current temperature divided by the starting temperature. Each variable in χ is then limited by the upper and lower bounds we previously set. The probability of any one variable becoming mutated is 20%. This level of mutation was selected so that neighbors would be different from the seed but not radically different. The MATLAB code for the neighbor generating function, *mutateSA()*, is provided in Appendix A.
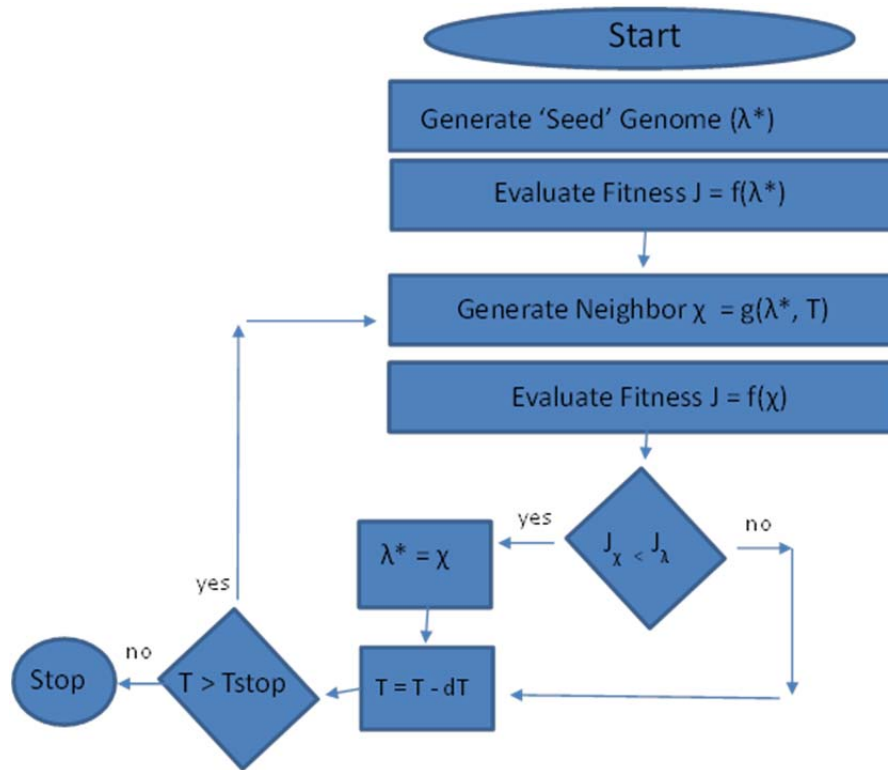
Figure 29. Simulated Annealing Process Flow Chart

The cost performance results of simulated annealing optimization are displayed in Figure 30. These results are not impressive. Simulated annealing yielded approximately two-tenths of a decade in stored energy reduction over 3200 iterations of the algorithm. This poor performance may be due to selecting neighbors that are too similar to the seed value, or the algorithm may have been trapped in a local minimum. Inspection of Figure 30 shows that cost reduction had essentially stalled, so further iterations would not have improved performance. Changing the cooling profile of Figure 31 would likely not have much impact either, since the local minimum appears to have been found long before the cooling period. After several runs of simulated annealing, the performance of the routine did not produce results superior to a few manual iterations by a human operator. Dissatisfaction with simulated annealing led to an exploration of genetic algorithm as an alternate optimization tool.
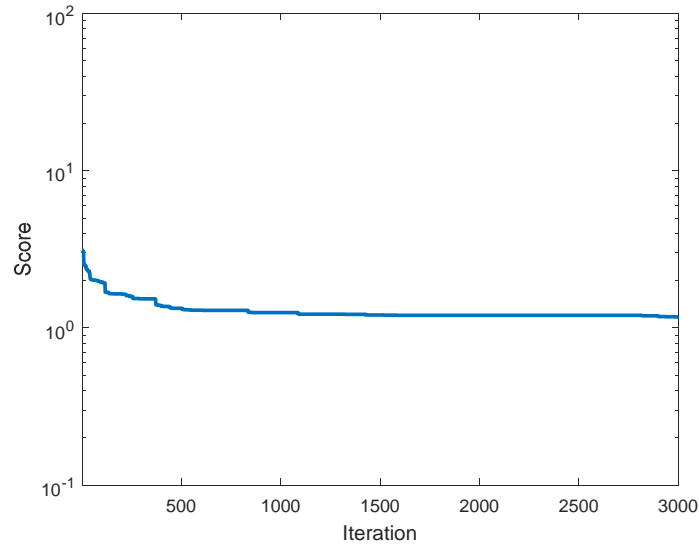
Figure 30. Simulated Annealing Cost Performance
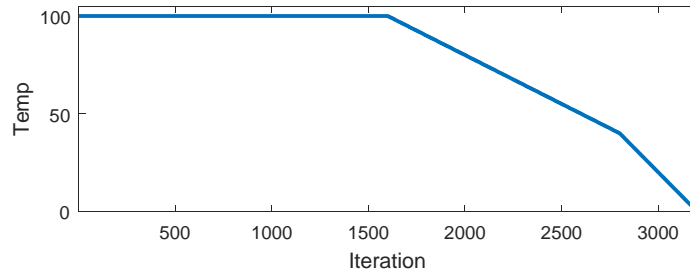
Figure 31. Simulated Annealing Temperature Profile

E.   GENETIC ALGORITHM

Genetic algorithm is a stochastic optimization algorithm that mimics biological reproduction processes. In genetic algorithm, an initial population is assembled from randomly generated points in the available design space. Each of these points in the design space is called a genome. Each genome is then evaluated according to a fitness function, just as we did with simulated annealing. In the next step, some fraction of the genomes in the generation is culled, leaving the best performing genomes in the generation to "breed" and produce the next generation. A certain amount of mutation is injected into the genomes of the next generation in order to produce new features not

67

present in the previous generation. Finally, the process is repeated until some termination criterion is reached. As before, the termination criterion could be a certain number of generations or a performance measure [41].

For our implementation in [42], we chose a generation composed of 16 genomes. We then refine the population over 200 generations. These numbers were chosen so that genetic algorithm would evaluate 3200 candidate genomes just as the simulated annealing algorithm evaluated 3200 candidates. Each genome is a vector of Q and R matrix diagonals along with the capacitor values we wish to minimize. This genome is identical to the one used for simulated annealing. The fitness function used in the genetic algorithm optimization is Equation (5.3), the same as was used in simulated annealing. An additional feature of genetic algorithm is the concept of *The Elite*. The Elite is the genome with the best observed fitness. The Elite passes unchanged from one generation to the next. In this way, each generation maintains memory of the best genome. The genetic algorithm block diagram is illustrated in Figure 32.

Just as with simulated annealing, we have to be careful to choose an initial generation that has at least one candidate pass all of the discrete fitness criteria for dynamics and harmonic content. One member of the initial generation of genomes was the same exact seed used to begin the simulated annealing optimization. The remaining fifteen members of the generation were genomes where approximately 20% of individual traits were selected by a uniform random variable that spanned the entire range for that particular design variable. The remaining 80% of variables were set as identical to the seed genome. After each genome has been scored by the fitness function, we cull the worst half of the genomes. The remaining best performing genomes are paired with one another and their genome variables mixed to form children. Genome variables are chosen according to a uniform random variable for "mixing." The mixing vector M is an $N{\times}1$ vector of ones and zeros where N is the length of the genome. The mixing vector complement is $\bar{M} = 1 - M$. Children are created according to

$$
\begin{aligned}
Child_1 &= P_A M + P_B \bar{M} \\
Child_2 &= P_A \bar{M} + P_B M
\end{aligned}
, \tag{5.4}
$$

where P_A is the first parent and P_B is the second parent. In order to create a truly new generation, the four genomes (P_A, P_B, and the two children) are mutated according to a mutating function. If the elite genome is present in the pairing, the elite genome is not mutated. The mutating function used is very similar to the neighbor generating function used in simulated annealing. The major differences are that the variance is fixed at 0.5 rather than modulated by the global temperature variable.

Various combinations of population size and number of generations were tried. A population of eight members and 400 generations as well as a population of 32 members and 100 generations were also tried. Best results were obtained with the population of 16 members and 200 generations.

The genetic algorithm cost performance chart, Figure 33, shows that genetic algorithm vastly outperformed simulated annealing. While simulated annealing only achieved two-tenths of a decade in stored energy reduction, genetic algorithm achieved nearly two-and-a-half decades in cost reduction. Specifically, the lowest system stored energy requirement determined by simulated annealing was 1.176 MJ, while the genetic algorithm was able to find a configuration of only 343 kJ. The genetic algorithm optimized system requires only one-third the stored energy of the simulated annealing optimized system.
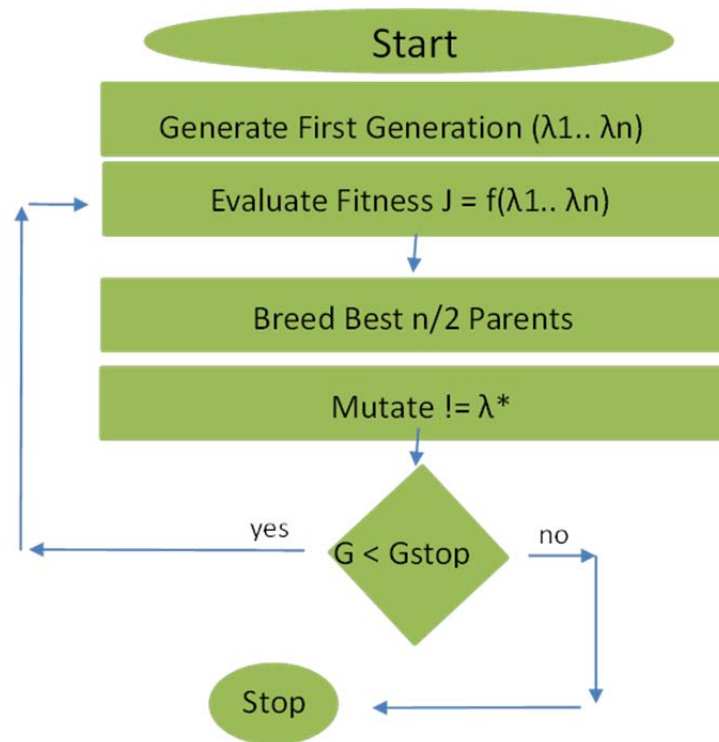
Figure 32. Genetic Algorithm Process Flow Chart
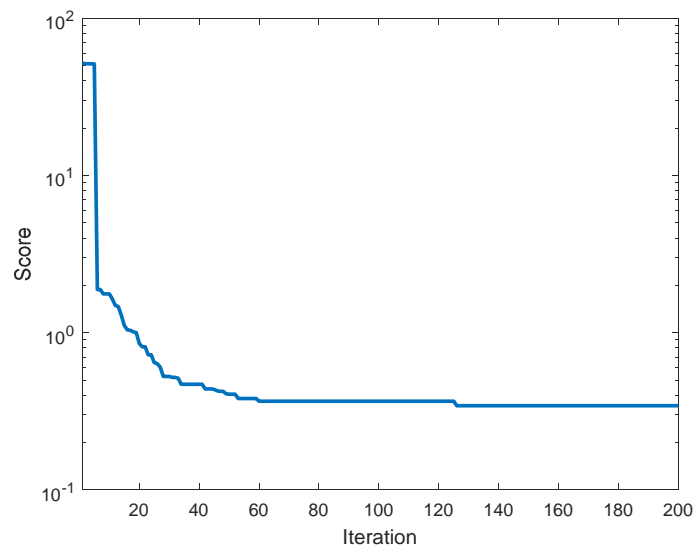
Figure 33. Genetic Algorithm Cost Performance

F.   COMPARISON TO STATE-FEEDBACK LINEARIZATION

This section explores how well the genetic algorithm-optimized LQR-SM controller compares to another controller. State-feedback linearization (LSF) is a non-linear control technique that is relatively straightforward to understand and implement. Several examples of LSF controlled systems exist in the literature and were explored in Chapter II of this work. Reference [22] forms the foundation for the LSF implementations used in this section.

To implement LSF, we first partition the circuit of Figure 12 into second-order subsystems. The first subsystem is composed of the four PGMs, the MVDC bus, and an ideal CPL. For this first subsystem, we make two key assumptions. The first key assumption is an identical ratio of PGM output resistance to PGM output inductance for each of the four PGMs. This assumption allows us to combine all four PGMs into one single virtual machine. The second key assumption is that all three of the load zones can be represented by a single ideal CPL. By doing so, we can use the analysis of [22] to design a second-order linear control law for the PGMs to stabilize the MVDC bus.

The remaining subsystems are in the load zones. By assuming that the MVDC voltage is constant, we can reduce each of the zones into an ideal voltage source, source inductance, filter capacitor, and ideal CPL. The three zones are now reduced to second-order systems. We can then design second-order control laws for the HESS currents to stabilize the zone voltages.

1.   PGM Subsystem

For the subsystem consisting of the four PGMs and the MVDC bus, the load zones are all approximated by a single CPL. The equivalent circuit model is shown in Figure 34.
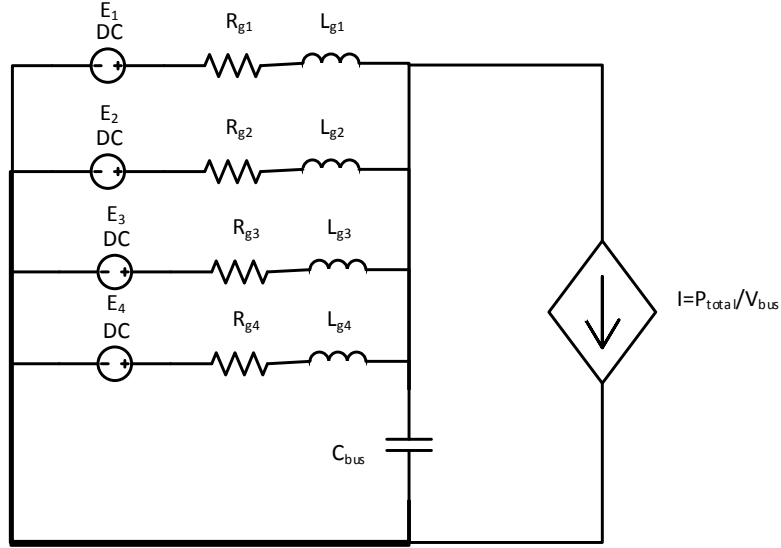
Figure 34. PGM Subsystem Circuit Diagram

To develop the control law for each of the PGM voltages, we start with the differential equations for the system in Figure 34, given as

$$\dot{I}_{g1} = \frac{1}{L_{g1}}\left(E_1 - R_{g1}I_{g1} - V_{bus}\right)$$

$$\dot{I}_{g2} = \frac{1}{L_{g2}}\left(E_2 - R_{g2}I_{g2} - V_{bus}\right)$$

$$\dot{I}_{g3} = \frac{1}{L_{g3}}\left(E_3 - R_{g3}I_{g3} - V_{bus}\right)$$

$$\dot{I}_{g4} = \frac{1}{L_{g4}}\left(E_4 - R_{g4}I_{g4} - V_{bus}\right)$$

$$\dot{V}_{bus} = \frac{1}{C_{bus}}\left(I_{g1} + I_{g2} + I_{g3} + I_{g4} - \frac{P}{V_{bus}}\right)$$ (5.4)

By taking the derivative of the fifth differential equation of Equation (5.4), we obtain

$$\ddot{V}_{bus} = \frac{1}{C_{bus}}\left(\sum_{i=1}^{4}\frac{1}{L_{gi}}(E_i - R_{gi}I_{gi} - V_{bus}) + \frac{P}{V_{bus}^2}\dot{V}_{bus}\right).$$ (5.5)

72

If we set the condition that the ratio of R_{gi} to L_{gi} for all PGMs are equal, we can further simplify Equation (5.5) to

$$\ddot{V}_{bus} = \frac{1}{C_{bus}}\left(\sum_{i=1}^{4}\frac{E_i}{L_{gi}} - \tau\sum_{i=1}^{4}I_{gi} - \sum_{i=1}^{4}\frac{1}{L_{gi}}V_{bus} + \frac{P}{V_{bus}^2}\dot{V}_{bus}\right)$$

$$\ddot{V}_{bus} = \frac{1}{C_{bus}}\left(\sum_{i=1}^{4}\frac{E_i}{L_{gi}} - \tau\left(C_{bus}\dot{V}_{bus} - \frac{P}{V_{bus}}\right) - \frac{1}{L_{eq}}V_{bus} + \frac{P}{V_{bus}^2}\dot{V}_{bus}\right)$$

$$\ddot{V}_{bus} = \sum_{i=1}^{4}\frac{E_i}{C_{bus}L_{gi}} - \tau\dot{V}_{bus} - \frac{\tau P}{C_{bus}V_{bus}} - \frac{1}{C_{bus}L_{eq}}V_{bus} + \frac{P}{C_{bus}V_{bus}^2}\dot{V}_{bus}$$

$$\ddot{V}_{bus} + \tau\dot{V}_{bus} + \frac{1}{C_{bus}L_{eq}}V_{bus} = \sum_{i=1}^{4}\frac{E_i}{C_{bus}L_{gi}} - \frac{\tau P}{C_{bus}V_{bus}} + \frac{P}{C_{bus}V_{bus}^2}\dot{V}_{bus}$$

(5.6)

where τ is the ratio of R_{gi} to L_{gi} and L_{eq} is the equivalent parallel inductance.

Suppose we set each of the PGM voltages to

$$E_i = V_{ref} + S_i C_{bus} L_{gi}\left(K_1(V_{bus} - V_{ref}) + K_2\dot{V}_{bus} + \frac{\tau P}{C_{bus}V_{bus}} - \frac{P}{C_{bus}V_{bus}^2}\dot{V}_{bus}\right) \qquad (5.7)$$

The parameter S_i is a sharing or proportioning constant that allows each PGM to share current in proportion to its rating. The sum of all sharing constants must add up to one. The K_1 and K_2 constant are the gains of a proportional and differential controller. Notice that Equation (5.7) has both a linear part and a non-linear part. The non-linear part of Equation (5.7) cancels out the non-linear part of Equation (5.6), linearizing the system. By substituting Equation (5.7) into Equation (5.6), we get

$$\ddot{V}_{bus} + (\tau - K_2)\dot{V}_{bus} + \left(\frac{1}{C_{bus}Leq} - K_1\right)(V_{bus} - V_{ref}) = 0 \qquad (5.8)$$

as the transfer function for the linearized system. We then choose the K_1 and K_2 variables to affect pole placement. Using the standard form of

$$s^2 + 2\zeta\omega_0 s + \omega_0^2 = 0 \qquad (5.9)$$

for a second-order linear differential equation, we set K_1 and K_2 according to

73

$$K_1 = \frac{1}{C_{bus}L_{eq}} - \omega_0^2$$

$$K_2 = \tau - 2\zeta\omega_0 \tag{5.10}$$

to obtain the desired system dynamics.

2.   Zone Subsystems

To perform control of the zones, we make the simplifying assumption that the zone buck voltage is constant. This is actually a fairly accurate assumption considering that we limit MVDC voltage fluctuation to less than ±10%. The simplified zone circuit diagram is shown in Figure 35. As before, we start with the differential equations which result from the subsystem circuit model, given as

$$\dot{I}_{bx} = -\frac{(V_{bx} - V_{refx})}{L_{bx}}$$

$$\dot{V}_{bx} = \frac{1}{C_{bx}}(I_{bx} - \frac{P_x}{V_{bx}} + I_{sx}) \tag{5.11}$$
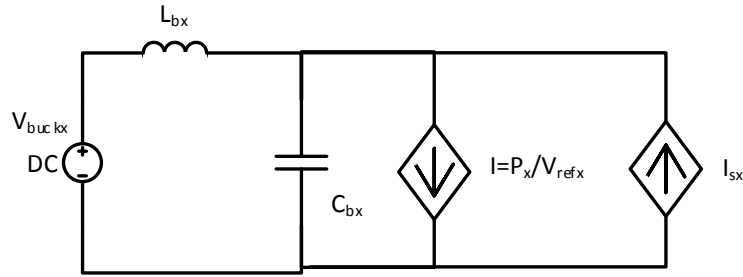
Figure 35. Zone Subsystem Circuit Model

Next, we take the time derivative of the second equation of Equation (5.11), then substitute that into the first equation of Equation (5.11), to get

74

$$\ddot{V}_{bx} = \frac{1}{C_{bx}}(\dot{I}_{bx} + \frac{P_x}{V_{bx}^2}\dot{V}_{bx} + \dot{I}_{sx})$$

$$\ddot{V}_{bx} = \frac{1}{C_{bx}}(-\frac{(V_{bx} - V_{refx})}{L_{bx}} + \frac{P_x}{V_{bx}^2}\dot{V}_{bx} + \dot{I}_{sx})$$

$$\ddot{V}_{bx} - \frac{P_x}{C_{bx}V_{bx}^2}\dot{V}_{bx} + \frac{(V_{bx} - V_{refx})}{C_{bx}L_{bx}} = \dot{I}_{sx} \qquad\qquad (5.12)$$

If we then posit that the HESS current I_{sx} is equal to

$$I_{sx} = K_3(V_{bx} - V_{refx}) + K_4 \int (V_{bx} - V_{refx}) \, dt \qquad\qquad (5.13)$$

we get

$$\ddot{V}_{bx} - \frac{P_x}{C_{bx}V_{bx}^2}\dot{V}_{bx} + \frac{(V_{bx} - V_{refx})}{C_{bx}L_{bx}} = K_3\dot{V}_{bx} + K_4(V_{bx} - V_{refx})$$

$$\ddot{V}_{bx} - (\frac{P_x}{C_{bx}V_{bx}^2} + K_3)\dot{V}_{bx} + (\frac{1}{C_{bx}L_{bx}} - K_4)(V_{bx} - V_{refx}) = 0 \qquad\qquad (5.14)$$

In this case, K_3 and K_4 are the respective gains of a proportional and integral controller. We then choose the K_3 and K_4 variables to affect pole placement. Using the standard form of Equation (5.9), we choose K_3 and K_4 according to

$$K_3 = -\frac{P_x}{C_{bx}V_{bx}^2} - 2\zeta\omega_0$$

$$K_4 = \frac{1}{C_{bx}L_{bx}} - \omega_0^2 \qquad\qquad (5.15)$$

to obtain the desired dynamic responses.

3.   LSF Parameter Optimization

In order for the LSF controllers to compete on even footing with the LQR-SM controller, the LSF design parameters were also optimized using genetic algorithm. The identical genetic algorithm routine was used to optimize the LSF controllers as well as the capacitors we wish to minimize. The LSF controlled system has three separate controllers that are simultaneously optimized: one for the PGM subsystem and the two

zones with HESS. Each controller needs to have the damping factor ζ as well as the natural frequency ω_0 selected. That makes six design variables for the controllers. The seven capacitors we wish to minimize increase the total number of design variables to thirteen. The design variables are selected by genetic algorithm with a population of 16 and refined over 200 generations. The fitness function Equation (5.3) is used to score each configuration.

4. Results

The results presented here are for the design transient stepping from 50% power to 100% power then back to 50% power again. The circuit model is given by Figure 12 with component values given in Table 4 and Table 5. The circuit parameter values obtained for the LSF based control scheme are given by Table 4. The genetic algorithm was used to optimize for minimum stored energy. The same optimization was performed for the LQR-SM rotating R controller, for which circuit values are listed in Table 5. The capacitance values differ between the two controllers, but all other circuit parameters remain identical. The damping factors and natural frequencies used by the LSF controllers are also contained in Table 4. Of particular note, the damping factors were restricted to values between 0.1 and 1.0, while the natural frequencies were restricted from 10.0 Hz to 8.0 kHz. From Table 4, we see that some of these values were optimized at the limits of their respective ranges. Expanding the range of values may provide better results for the LSF controllers.

Table 4.   LSF Component Values

C_{bus}	333 μF	C_{z1}	2.46 μF	C_{b1}	75 mF
C_{d1}	20.3 μF	C_{z2}	3.69 μF	C_{b2}	0.7 mF
C_{d2}	0.16 μF	C_{z3}	6.16 μF	C_{b3}	2.6 mF
C_{d3}	12.7 μF				
ς_{MVDC}	0.72	ς_{Zone1}	0.87	ς_{Zone2}	64
ω_{MVDC}	474	ω_{Zone1}	8000	ω_{Zone2}	7628

Table 5. LQR-SM Component Values

R_{g1}	0.25 Ω	R_{z2}	2.20 mΩ	R_{d3}	10.0 Ω
R_{g2}	0.30 Ω	R_{z3}	1.30 mΩ	C_{d1}	0.24 µF
R_{g3}	0.26 Ω	L_{z1}	70.5 µH	C_{d2}	66 µF
R_{g4}	0.32 Ω	L_{z2}	47.0 µH	C_{d3}	65 µF
L_{g1}	2.00 mH	L_{z3}	28.2 µH	L_{b1}	30.6 µH
L_{g2}	1.80 mH	C_{z1}	2.46 µF	L_{b2}	1.8 mH
L_{g3}	1.95 mH	C_{z2}	3.69 µF	L_{b3}	298 µH
L_{g4}	1.71 mH	C_{z3}	6.15 µF	C_{b1}	75 mF
C_{bus}	1.02 µF	R_{d1}	10 Ω	C_{b2}	0.7 mF
R_{z1}	3.30 mΩ	R_{d2}	10 Ω	C_{b3}	1.9 mF

The voltage transient comparison we see in Figure 36 illustrates that the LSF control scheme provides a very classical type response with a decaying oscillation. In contrast, the LQR-SM response has relatively few oscillations. Except for Zone #2, both controllers have very similar overshoot and undershoot values as well as very similar settling times. This is to be expected: the genetic algorithm selected for these traits. In Figure 36a, we see a classical critically-damped response from the LSF controller while the LQR-SM controller sharply peaks on undershoot. The MVDC bus voltage swells to maximum overshoot before quickly settling to steady-state. In Figure 36b, we notice the LSF controlled system has very little overshoot. On the other hand, LQR-SM uses the full overshoot range. In Figure 36c, both controllers provide an underdamped response, but the LQR-SM controller returns to steady-state more quickly than the LSF controller. The Zone #3 voltage response in Figure 36d is not regulated by any controller. In both the LQR-SM and LSF cases, Zone #3 voltage mostly tracks with MVDC bus voltage fluctuations.
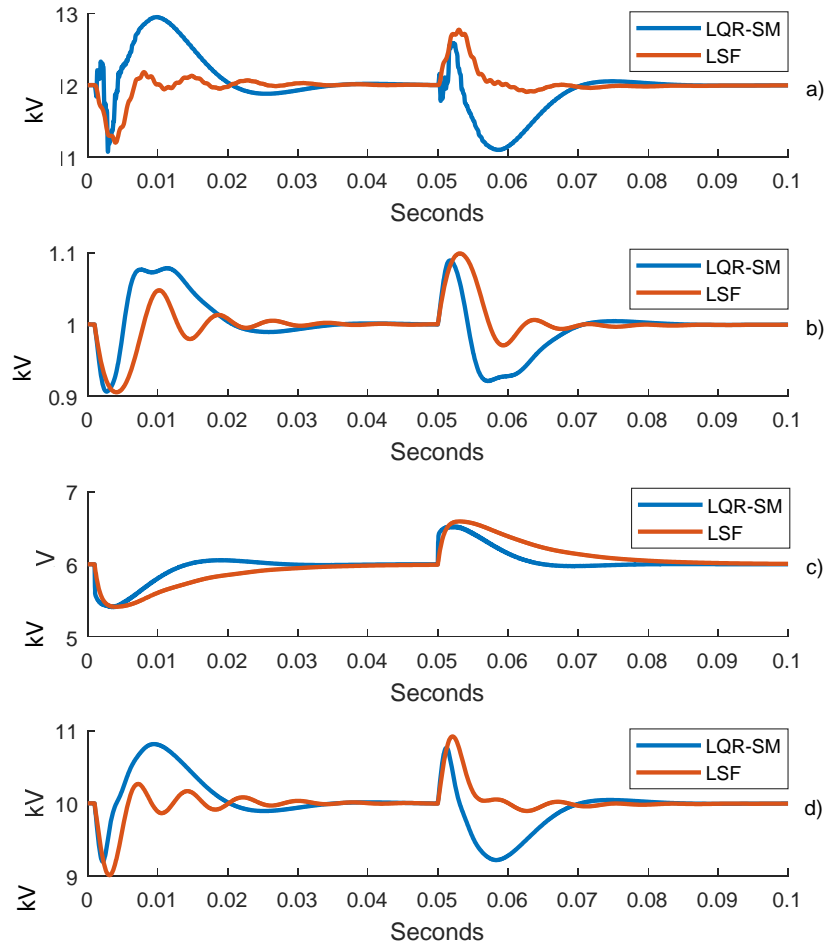
Figure 36. Optimized Controller Voltage Transients for MVDC Bus (a),
Zone #1 (b), Zone #2 (c), and Zone #3 (d)

The two controllers behave quite differently, especially in regard to PGM voltage control. The LQR-SM controller, upon sensing the step-change in loading, responds by rapidly increasing PGM voltages, shown in Figure 37a. This anticipatory surge in PGM voltage raises the PGM current quickly, allowing the MVDC bus to better maintain regulation while supplying the load zones. This rapid rise in PGM current better supplies the load current so that less current needs to be supplied by capacitors. Less demand for capacitor current allows capacitors to be sized smaller while still maintaining bus regulation.

In contrast, the LSF controller is more reactive than proactive. The LSF controller can only respond to correct a drop in MVDC bus voltage. The LSF controller, which is decentralized, cannot forestall voltage swings through anticipatory shifts in PGM current. This implies that the LSF controlled system requires larger capacitors to effectively regulate the MVDC bus.

The LSF controller PGM voltages are shown in Figure 37b. Notice how the range of values of the LSF controlled PGMs is much narrower than the range of values used by the LQR-SM controller in Figure 37a. The LSF controller does not require a large gain range for the PGM DC-DC converters, while the LQR-SM controller utilizes a much wide gain range. This result implies that control saturation is more likely with the LQR-SM controller.
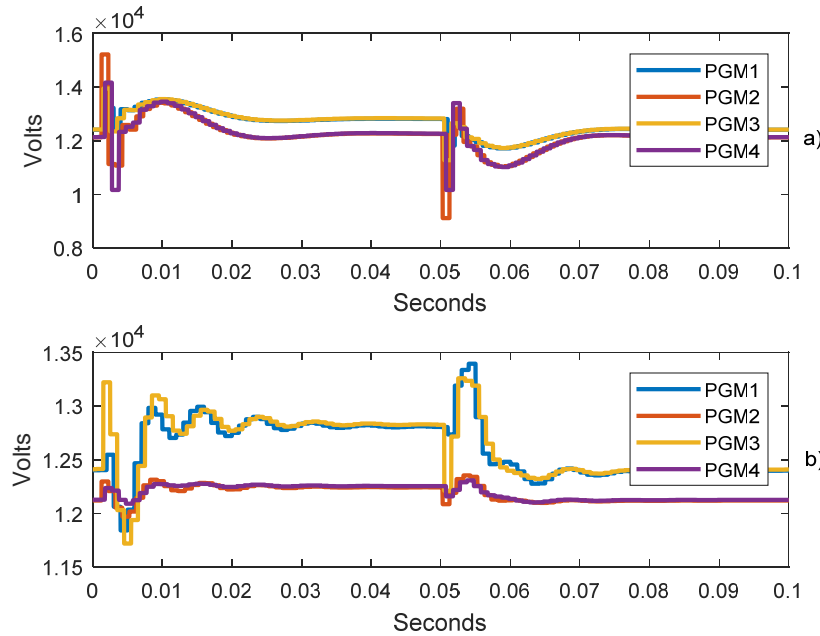
Figure 37. PGM Commanded Voltage for LQR-SM (a), and LSF (b)

Now that we have investigated the transient voltage response, let us consider which system is more efficient at utilizing stored energy. In Figure 38a, we see the transient curve for power delivered by HESS #1 and Figure 38b shows the transient curve

for HESS #2. In both cases, the LSF controller requires greater peak energy from the HESS. This appears to be due to the LSF controller simply taking longer to overcome the transient. The LQR-SM controller more rapidly returns zone bus voltages to steady-state values, thereby saving significant amounts of HESS energy. These points are illustrated in Figure 38 and Figure 39. The LQR-SM controller HESS #1 energy peak is about 69% of the LSF controller HESS #1 energy peak. The LQR-SM HESS #2 energy peak is about 51% of that used by the LSF controller. Aside from the energy savings in HESS #1 and HESS #2 control effort, the capacitors used by the LQR-SM controlled system are smaller as well. Overall, the LQR-SM controlled system has a stored energy requirement of only 61% of the LSF controlled system (339 kJ for LQR-SM versus 555 kJ for LSF). The differences in performance may be attributable to the fact that the LQR-SM controller is a centralized controller with full system knowledge while the LSF controllers are decentralized. The LQR-SM controller can synergize all of the available inputs in order to regulate the system. In contrast, each LSF controller is only acting to regulate its local bus. The lack of harmony between controllers contributes to their inefficiency.
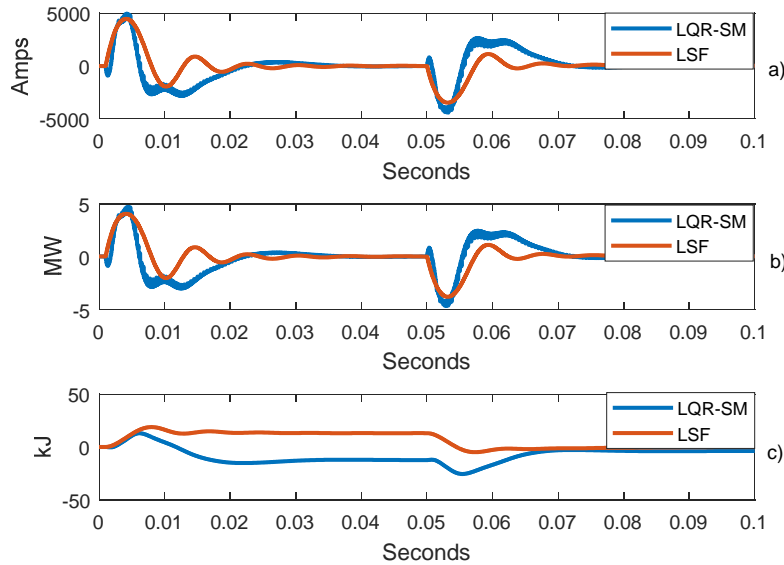
Figure 38. HESS #1 Transient Response for Current (a), Power (b),
and Energy (c)
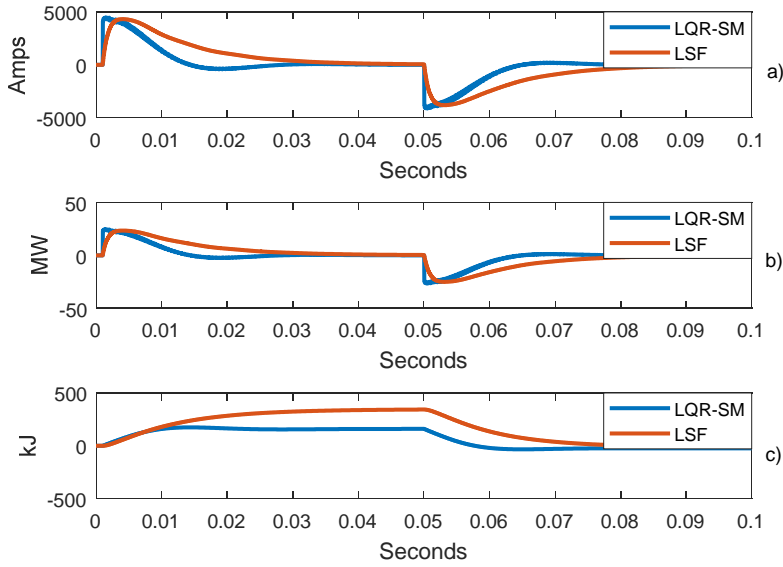
Figure 39. HESS #2 Transient Response for Current (a), Power (b),
and Energy (c)

G.   SUMMARY

In this chapter, we explored stochastic search algorithms as a method to design LQR-SM controllers. Both simulated annealing and the genetic algorithm were applied to the task of designing an LQR-SM controller while simultaneously minimizing total system energy storage.

The simulated annealing algorithm did not perform as well as the genetic algorithm for our example system. The genetic algorithm, by contrast, provided a system and controller that met dynamic and harmonic design goals with much less stored energy than was required by the simulated annealing optimization.

In producing an optimized LQR-SM controller and system, we wished to compare the performance of the LQR-SM controller to a benchmark controller. We selected LSF as the benchmark due to its popularity in the literature and relative ease of understanding and implementation. In order to produce the fairest possible comparison, the LSF controller was also optimized by the genetic algorithm using the exact same population size, number of generations, and fitness function. By comparing the minimum stored

81

energy configurations of both LQR-SM and LSF, we found that the LQR-SM controlled configuration required only 61% of the stored energy required by LSF to stabilize the transient from 50% power to 100% and back to 50% power.

The LQR-SM controller is superior to the de-centralized LSF controllers. LQR-SM requires fewer simplifications and assumptions than the LSF controller. LQR-SM, as a centralized approach, is able to harmonize all available inputs to provide optimal system-wide regulation of all buses. Unlike LSF, there is no possibility of two different de-centralized regulators working at cross purposes (i.e., fighting controllers). Overall, the LQR-SM controller provides excellent bus regulation while requiring smaller capacitors and lower capacity HESS as compared to an LSF controlled system. The drawback of the LQR-SM controller is greater computational complexity due to the requirement to solve the Riccati equation for large matrices.

VI.   PERFORMANCE ENHANCEMENT

One of the weaknesses of the LQR-SM controller is that solving the Riccati equation in real time is computationally expensive. For non-adaptive systems, LQR controllers are typically solved off-line, and the feedback gain matrix is fixed. By shifting from a fixed gain matrix to an adaptive gain matrix, we introduce the computational expense of solving the Riccati equation every switching cycle. Using our knowledge of the MVDC distribution system we have been studying, we explore reduced-order controllers and the design trade-offs imposed by them. Since our system is not fully-controllable, we have many uncontrolled states. Our system is nevertheless stabilizable, so these uncontrolled states are stable. We can develop controller models that omit or consolidate the uncontrolled states, then explore the effects these reduced-fidelity models have on controller performance. The reduced-order controllers are then applied to the full twentieth-order circuit model and optimized with the genetic algorithm as previously described.

A.   ADAPTIVE 17TH-ORDER LQR-SM CONTROLLER

When examining the results of the LQR-SM controlled system, we can see that the MVDC bus voltage, the voltage at the input to the zone buck converters (V_{z1}, V_{z2}, V_{z3}), and the voltages on the filter capacitors (V_{d1}, V_{d2}, V_{d3}), are nearly identical. If we make the assumption that the filter damper resistor R_{dx} is negligible, we can combine the cabling parasitic capacitances with the filter capacitances. This simplification reduces the order of the controller from twentieth order to seventeenth order. The seventeenth-order circuit model is shown in Figure 40 with the regions of simplification circled in red.

Just as with the adaptive twentieth-order LQR-SM controller, the adaptive seventeenth-order LQR-SM controller is optimized to meet our dynamic and harmonic specifications with minimal stored energy using identical methods to those described for adaptive twentieth-order LQR-SM.
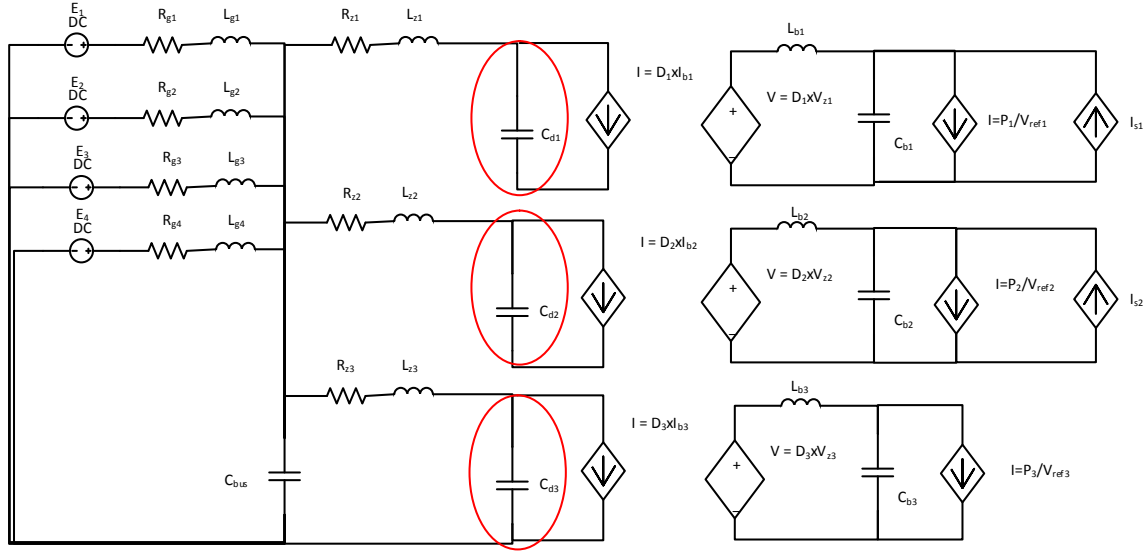
Figure 40. Seventeenth-Order Controller Circuit Model

B.   ADAPTIVE 11TH-ORDER LQR-SM CONTROLLER

As with the seventeenth-order controller, we again note that MVDC voltage and the zone buck input voltages are virtually identical. From this, we conclude that the cabling parasitic resistances and inductances (R_{zx} and L_{zx}) are negligible. By ignoring the impedances of the zone cables, we can further simplify the controller to eleventh-order. As with the seventeenth-order controller, we also ignore the filter damper resistance. This allows us to combine the PGM output capacitors with the filter capacitors to form a single equivalent capacitor. The eleventh-order circuit model is shown in Figure 41 with the regions of simplification circled in red.

Just as with the adaptive twentieth-order and seventeenth-order LQR-SM controllers, the adaptive eleventh-order LQR-SM controller is optimized to meet our dynamic and harmonic specifications with minimal stored energy using identical methods to those described for adaptive twentieth-order LQR-SM.
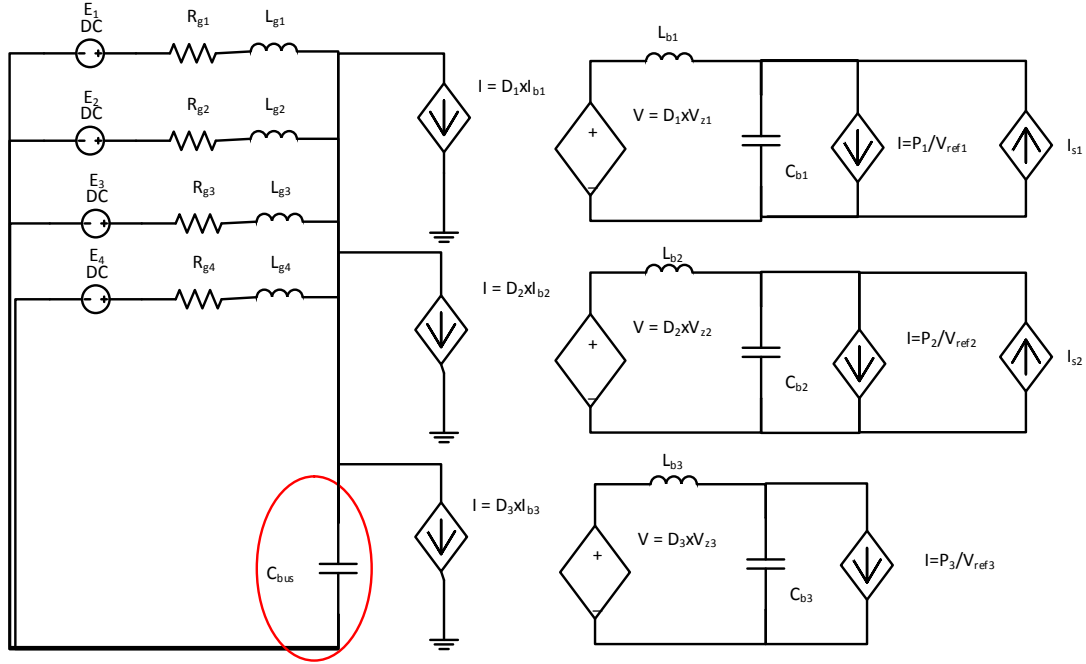
Figure 41. Eleventh-Order Controller Circuit Model

C. NON-ADAPTIVE 20TH-ORDER CONTROLLER

Having reduced the order of the controllers to the maximum extent, we next considered what performance trade-off might occur by calculating the feedback gains off-line. For this experiment, we used a non-adaptive LQR-SM controller. Rather than estimating power in real time, we linearized the system about steady-state 100% power operation and calculated the feedback gain matrices for each switching combination. In our case, we have four PGMs and two HESS. Just as with the adaptive LQR-SM controllers, we have 16 sub-cycles in a super-cycle and five different switching input combinations. In the first sub-cycle, PGM #1 is switching along with HESS #1 and HESS #2. The next three sub-cycles have just the two HESS updating. The fifth sub-cycle has PGM #2 and the two HESS switching followed by three sub-cycles of just the HESS switching, and so on. This results in five distinct gain matrices. One gain matrix is used when switching PGM #1 and the two HESS, the second for switching PGM #2 and the two HESS, the third for switching PGM #3 and the two HESS, the fourth for switching PGM #4 and both HESS, and fifth gain matrix is for switching the two HESS and no

PGMS. Once the gain matrices are computed offline, they are stored in memory and called upon during the appropriate sub-cycle to compute the PGM and HESS input values.

Just as with the adaptive LQR-SM controller, the non-adaptive LQR-SM controller was optimized to meet our dynamic and harmonic specifications with minimal stored energy using identical methods to those described for adaptive twentieth-order LQR-SM.

D.   RESULTS

1.   Minimum Energy Controllers

a.   *Calculation Speed*

To compare the computational load of each controller, the different control strategies were implemented in MATLAB simulations. While inputs to the PGMs and HESS were recalculated every 62.5 microseconds, the simulation updated the differential equations every 0.1 microseconds. The figures displayed are produced by sampling the inputs and state variables at 40.0 kHz. The data streams were down-sampled due to limitations in computer memory. The time duration of the simulations was recorded using the MATLAB *tic* and *toc* functions.

The different controllers' speed performance improved by reducing the order of the controller, as expected. The non-adaptive controller was obviously the fastest, completing the 0.1-s simulation trial in 12.8-s. Since the non-adaptive controller only computes the Riccati equation once off-line, there are no Riccati equation computations included in that 12.8-s period. We can regard this simulation time as the minimum amount of time that our MATLAB script can run on the available desktop computers. The next fastest controller was the eleventh-order controller, which clocked a 13.6-s simulation run. This 0.8-s difference represents 1600 Riccati equation computations for an average of 500 microseconds of computing time for each eleventh-order Riccati solution. The seventeenth-order controller was the next fastest at 13.7-s. The jump from eleventh-order to seventeenth-order only added an additional 0.1 s of computing time.

This averages to 563 microseconds per Riccati solution. The twentieth-order controller was, as expected, the slowest at 14.2 s. This averages to 875 microseconds per Riccati solution. Calculation results are displayed in Table 6.

Table 6.   LQR-SM Controller Computation Times

Controller	Non-adaptive	11th-Order	17th Order	20th-Order
Seconds	12.8	13.6	13.7	14.2

b.   Energy Storage Requirement

The next basis for comparison is determining if there is a performance trade-off with respect to energy-storage requirements with reduced-order controllers. The hypothesis going into this comparison was that reducing the fidelity of the controller models would result in reduced "optimality" from the controllers; therefore, each reduction in fidelity of the controllers would result in an energy storage cost penalty. In Figure 42, we compare the genetic algorithm cost performance of each of the different types of controllers through 200 generations of optimization.

The LSF controller is the worst performer of the group and is included for the sake of having a baseline. The LSF controller had an energy storage requirement of 554.9 kJ. The three adaptive controllers had almost no differences in their performances. The eleventh-order, seventeenth-order, and twentieth-order controllers produced minimum energy storage requirements of 331.5 kJ, 337.0 kJ, and 331.6 kJ, respectively. The hypothesis that a loss of controller fidelity would result in reduction in optimality is false. The reason for the parity in performance between the adaptive controllers is most attributable to the fact that the states that were eliminated from the controller model were uncontrollable states. Although the voltages and currents eliminated from the controller model may have a minor effect on the remaining states, an examination of graphs of V_{bus}, V_{z1}, V_{z2}, V_{z3}, V_{d1}, V_{d2}, and V_{d3} shows that these voltages are nearly identical. Eliminating impedances that had very minor effects on the circuit resulted in only minor effects on the controller.

The big surprise from this comparison is that the non-adaptive controller was optimized to a minimum energy storage requirement of only 306.9 kJ. The non-adaptive controller may have the stored energy advantage for a few reasons. The first reason is that the non-adaptive controller is designed for 100% power, so it does not have to delay by a super-cycle before all of the inputs are adjusted for the power jump. Step changes in loading below 100% power level may yield a different outcome. The second reason could be that the genetic algorithm located a more optimal result for the non-adaptive controller. The genetic algorithm has a great deal of randomness built into it. This particular optimization of the non-adaptive controller may have just been lucky with regard to receiving the right randomizations and mutations to produce an excellent result. Our results with the genetic algorithm have shown that all trials produce roughly equivalent scores, but there is no convergence to a single set of parameters. The genetic algorithm gives us many "very good" solutions but does not guarantee the "best" solution.
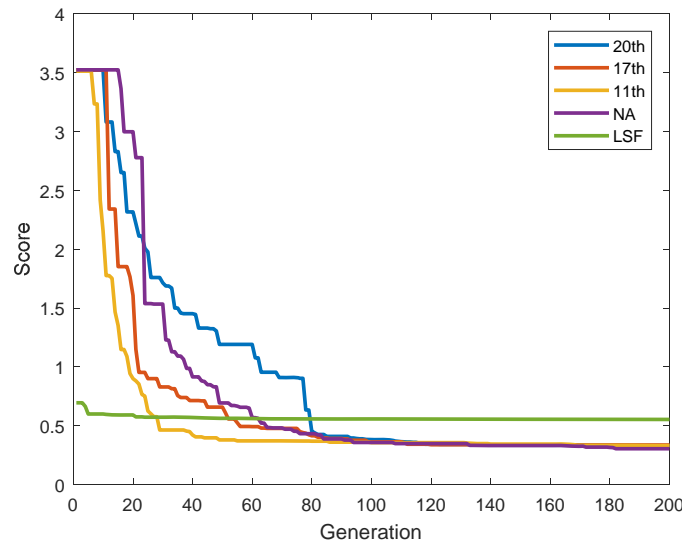
Figure 42. Comparison of Controller Genetic Algorithm Fitness

c. *Region of Attraction*

The ROAs were calculated using the same search technique used in Chapter IV. The circuits were established at 100% power and the zonal voltages were disturbed in the zonal I-V planes. If the circuit returned to steady-state operation without allowing bus voltage to dip below 0.0 V, without experiencing a Riccati solution error, and without exceeding a 40.0-ms settling time, the disturbance was evaluated as within the ROA. The goal is to understand what trade-offs can be expected between the different controllers.

In the minimum energy configuration, each circuit is optimized for minimum stored energy requirement. This means that each of these different controllers has unique capacitance values for C_{bus}, C_{d1}, C_{d2}, C_{d3}, C_{b1}, C_{b2}, and C_{b3}. The differences in capacitance values may play a factor in the size of the ROAs. The different capacitance values for each minimum energy configuration are recorded in Table 7.

Table 7.　Minimum Energy Configuration Capacitance Values

Capacitor	C_{bus}	C_{d1}	C_{d2}	C_{d3}	C_{z1}	C_{z2}	C_{z3}	C_{b1}	C_{b2}	C_{b3}
20th	5.4 μF	0.1 μF	0.3 μF	20 μF	2.5 μF	3.7 μF	6.2 μF	78 mF	0.7 mF	2.2 mF
17th	3.5 μF	18 μF	7.8 μF	8.0 μF	2.5 μF	3.7 μF	6.2 μF	75 mF	0.7 mF	2.2 mF
11th	3.5 μF	4.6 μF	2.6 μF	0.2 μF	2.5 μF	3.7 μF	6.2 μF	75 mF	0.8 mF	2.2 mF
NA	3.2 μF	0.1 μF	45 μF	1.1 μF	2.5 μF	3.7 μF	6.2 μF	75 mF	0.7 mF	1.4 mF
LSF	333 μF	20 μF	0.2 μF	13 μF	2.5 μF	3.7 μF	6.2 μF	75 mF	0.7 mF	2.6 mF

The Zone #1 ROAs are displayed in Figure 43. Immediately, we see that the ROAs are quite different from one another. The three adaptive controllers have somewhat similar ROAs, with the twentieth-order adaptive controller having the smallest ROA. Again, the non-adaptive twentieth-order LQR-SM controller surprises us by having the largest ROA. Examining the Zone #1 buck filter capacitors for each configuration, we can see that all of the configurations have very similar C_{b1} values. All of the configurations use a C_{b1} of 75 mF except for the twentieth-order adaptive LQR-SM

controller, which uses 78 mF. Despite each of these configurations having nearly identical C_{b1} values, the Zone #1 ROAs are quite different. The differences may lie in factors such as controller feedback gains or systemic interactions.

Examining the Zone #2 ROA, we have a similar story as with Zone #1. Prior to examining each of the ROAs, one would have suspected that the higher fidelity controllers would have an advantage over the lower fidelity controllers or that larger zone buck capacitors had an advantage over smaller capacitors, but neither of these predictions are supported by the evidence. In the Zone #2 ROAs displayed in Figure 44, the three adaptive controllers each perform very similarly, while the non-adaptive controller has a noticeably larger ROA. Again, looking at the C_{b2} values from Table 7, we see that capacitor size had no discernable effect on ROA size. The controller action must be playing a significant role. Even though all controllers have been selected through the genetic algorithm for the same dynamics, the differences in feedback gains seem to have a profound effect on ROAs.
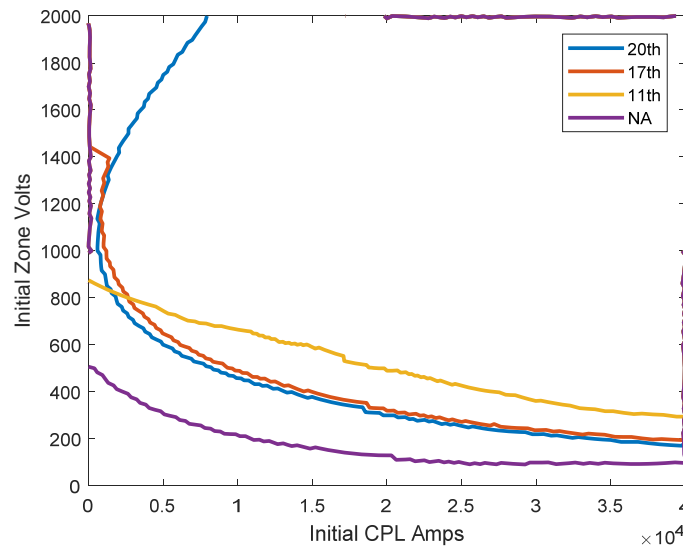
Figure 43. Zone #1 Minimum Energy ROAs
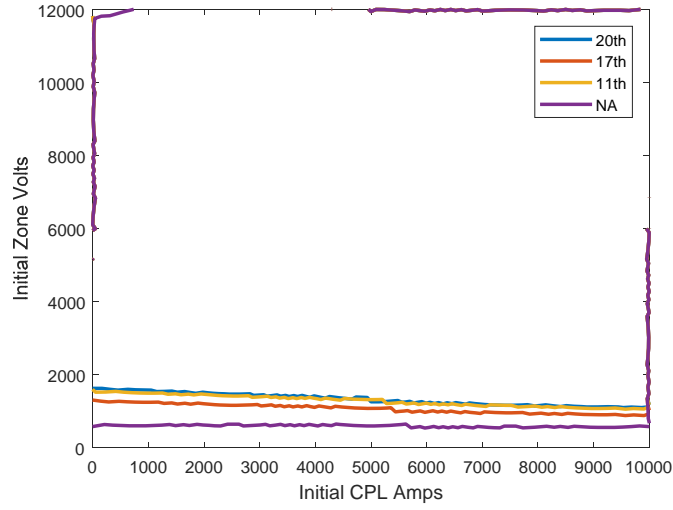
Figure 44. Zone #2 Minimum Energy ROAs

The Zone #3 ROAs, shown in Figure 45, are all quite similar. We expect that C_{b3} size will have more effect on Zone #3 than C_{b1} or C_{b2} had on their respective zones since Zone #3 does not have a HESS current source to influence dynamics. Despite all three of the adaptive controllers having identical C_{b3} values, there is a noticeable, but small, spread on Zone #3 ROAs. Even the non-adaptive controller, with its significantly smaller C_{b3} value, still has an ROA on par with the three adaptive controllers.
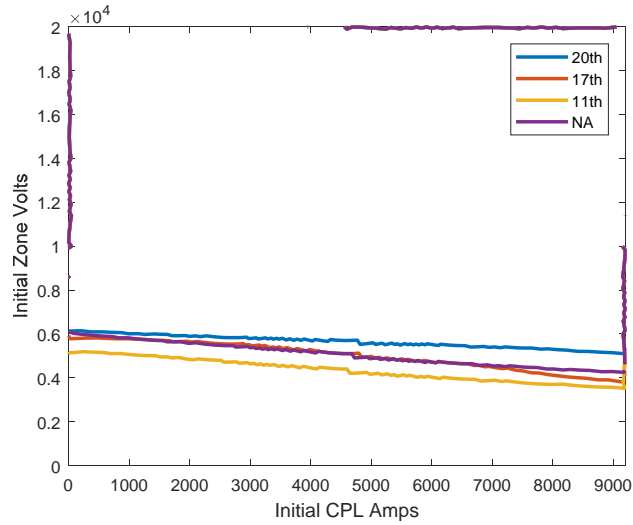
Figure 45. Zone #3 Minimum Energy ROAs

91

The LQR controller Q and R matrix diagonal values are recorded in Tables 8 and 9, respectively, for completeness. There is actually very little difference between the values in Table 9 for each controller. The differences in Q matrices are more varied, but in general, values are similar. It is interesting to note that each of these converters has a different solution for achieving the same result. All four of the LQR-SM controllers have been optimized by the genetic algorithm for 200 generations and each achieves a very similar final score. The genetic algorithm performance curves of Figure 42 might lead one to believe that these controllers have reached a global minimum or near global minimum. We have seen in actuality that each of the genetic algorithm routines has converged not on the unique "best solution" but on a set of "very good" solutions. Despite all of the solutions in the set of "very good" solutions meeting our minimum energy performance goals, there can be marked differences in the size and shape of ROAs created by each solution. This is another potential area for optimization. Unfortunately, generating a ROA takes approximately four hours of computation time, making optimization in this domain a costly endeavor.

Table 8.   Minimum Energy LQR Controller Q-values

Q-Penalty	20th	17th	11th	NA	Q-Penalty	20th	17th	11th	NA
I_{g1}	1	1.1	1	15	V_{z3}	1.2	1.5	-	1.6
I_{g2}	1	1.4	1	1.5	V_{d1}	1.1	-	-	2.9
I_{g3}	1	1.1	1	15	V_{d2}	2.2	-	-	2.2
I_{g4}	1	1.4.	1	1.5	V_{d3}	1.4	-	-	1.1
V_{bus}	1	1.6	2.2	1.2	I_{b1}	1	3.1	2.4	2.9
I_{z1}	1.1	6.8	-	1.3	I_{b2}	1.7	4.1	3.3	7.4
I_{z2}	4.9	6.7	-	1.7	I_{b3}	5.9	6.1	1	1.6
I_{z3}	1.9	1.9	-	24	V_{b1}	2.4	12	62	1
V_{z1}	1	3.3	-	2.9	V_{b2}	137	258	151	382
V_{z2}	1	2.6	-	1.2	V_{b3}	1	14	1.2	2.3

Table 9.   Minimum Energy LQR Controller R-values

Input R-Penalty	20th	17th	11th	NA
PGM #1/3	4	3	2.4	6.8
PGM #2/4	1	2	1	1
Unavailable	1000	1000	1000	1000
HESS#1	0.1	0.2	0.7	0.1
HESS#2	1	1	1	1

2.   Identical Configuration Controllers

Since the results of the minimum energy optimized controllers' ROAs variance seemed highly dependent on controller design parameters, in this section we investigate differences in controller performance when all controllers are using the same capacitor values and controller values. For this section, all four LQR-SM controller variants were optimized together using 200 generations of the genetic algorithm. In this optimization, all four controller variants were required to use the same capacitor values. To make the comparison even closer, each controller uses identical Q and R penalty values for state variables and inputs. For example, for all four controllers, the Q-matrix penalty on V_{b1} is identical. Even though the reduced-order controllers do not account for some state variables, at least there is parity between the controllers for all of the state variables they have in common.

Since nothing has changed with respect to computation of the respective controllers, the computation speeds of the identical configuration controllers are the same as those presented in the minimum energy configuration section.

The energy requirement for the identically configured controllers is in line with the results of the minimum energy configurations. The three adaptive controllers all require about the same amount of total stored energy, while the non-adaptive controller requires somewhat less energy. The twentieth-order controller required 345 kJ, the

seventeenth-order controller required 340 kJ, the eleventh-order controller needed 341 kJ, and the non-adaptive controller required only 329 kJ.

When examining the ROAs of the identically configured controllers, the three adaptive controllers had ROAs that almost perfectly overlapped. This fact is a major contrast to the amount of variability we saw in Figure 43. The non-adaptive controller has a more limiting ROA, as seen in Figure 46.
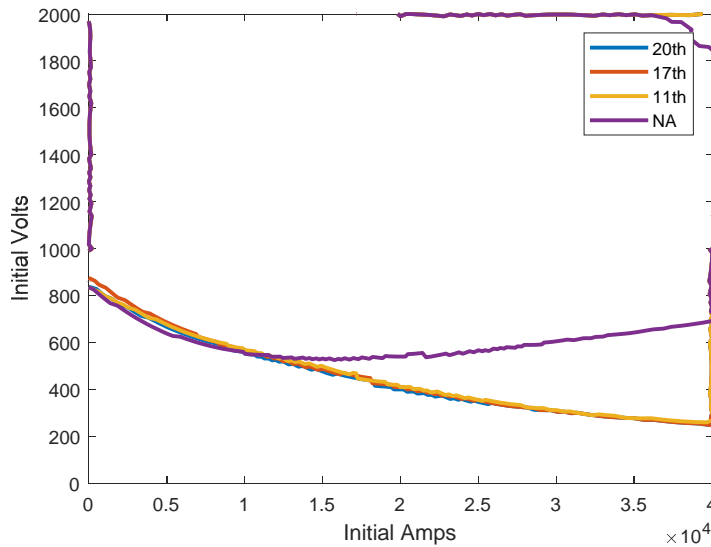
Figure 46. Zone #1 Identical Configuration ROAs

In Zone #2, we again see that the adaptive controllers all match quite closely with one another. The non-adaptive controller actually has a slightly larger ROA in this zone, though the difference is much less pronounced than in Zone #1. The Zone #2 ROAs for identical configuration are displayed in Figure 47. In Zone #3, the ROAs are all quite similar, though the non-adaptive controller's ROA is noticeably smaller than the adaptive controllers. The identical configuration Zone #3 ROAs are shown in Figure 48.

From these results, we can see that with identical Q and R matrix penalty values, the reduced-order controllers behave almost identically to one another except for improvements in computation speed. There is no measureable advantage to the higher-order adaptive controllers. The non-adaptive controller has shown, at least in the

realizations explored here, advantages in computation speed and stored energy requirement. The non-adaptive ROA size is an area of interest. In some cases, the non-adaptive controller has shown a larger ROA than the adaptive controllers but in other cases has shown a significantly smaller ROA.
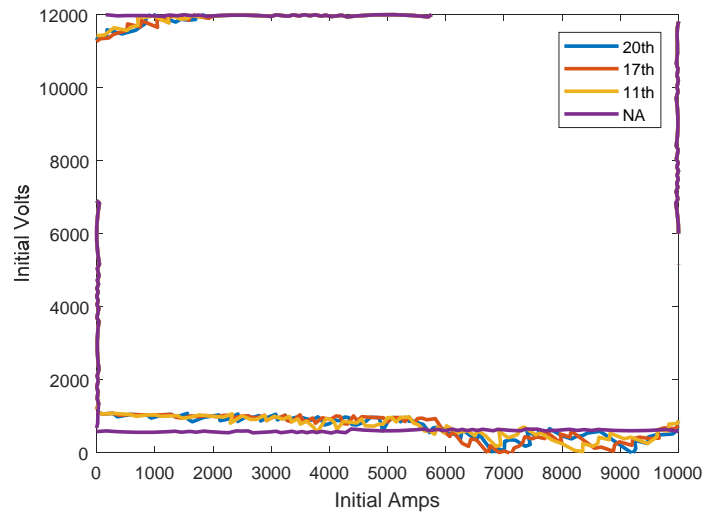
Figure 47. Zone #2 Identical Configuration ROAs
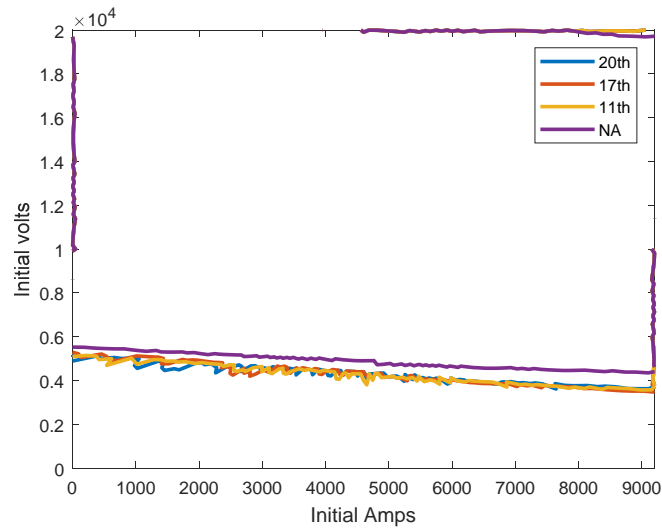
Figure 48. Zone #3 Identical Configuration ROAs

95

Identical configuration circuit component values are enumerated in Table 10. The Q-matrix values are listed in Table 11 with R values in Table 12.

Table 10.   Identical Configuration LQR Component Values

R_{g1}	0.25 Ω	R_{z2}	2.20 mΩ	R_{d3}	10 Ω
R_{g2}	0.30 Ω	R_{z3}	1.30 mΩ	C_{d1}	2.73 µF
R_{g3}	0.26 Ω	L_{z1}	70.5 µH	C_{d2}	38.2 µF
R_{g4}	0.32 Ω	L_{z2}	47.0 µH	C_{d3}	17.4 µF
L_{g1}	2.00 mH	L_{z3}	28.2 µH	L_{b1}	30.6 µH
L_{g2}	1.80 mH	C_{z1}	2.46 µF	L_{b2}	1.8 mH
L_{g3}	1.95 mH	C_{z2}	3.69 µF	L_{b3}	298 µH
L_{g4}	1.71 mH	C_{z3}	6.15 µF	C_{b1}	75 mF
C_{bus}	17.2 µF	R_{d1}	10 Ω	C_{b2}	0.7 mF
R_{z1}	3.30 mΩ	R_{d2}	10 Ω	C_{b3}	2.0 mF

Table 11.   Identical Configuration LQR Controller Q-Values

Q_{Ig1}	1.1	Q_{Iz1}	1.3	Q_{Vz3}	1.4	Q_{Ib2}	5.9
Q_{Ig2}	2.1	Q_{Iz2}	2.8	Q_{Vd1}	6.3	Q_{Ib3}	6.1
Q_{Ig3}	1.1	Q_{Iz3}	2.9	Q_{Vd2}	8.6	Q_{Vb1}	4.8
Q_{Ig4}	2.1	Q_{Vz1}	2.1	Q_{Vd3}	2.4	Q_{Vb2}	364
Q_{Vbus}	14.5	Q_{Vz2}	1.4	Q_{Ib1}	5.9	Q_{Vb3}	1.3

Table 12.   Identical Configuration LQR Controller R-Values

PGM #1/3	PGM #2/4	ESD #1	ESD #2	Rmax
4.4	1.9	0.9	1.0	1000

E.   SUMMARY

In this chapter, we explored various design trade-offs. The MVDC distribution system we model has several state variables that very closely track one another. These state variables arise due to modeling of bus impedances and input filters. If we assume

that input filter resistance values are negligible, we achieve a reduced-order model with only 17 state variables. If we also assume that bus impedances may be ignored, we can eliminate six more state variables to achieve an eleven state-variable model.

By reducing the order of our model through elimination of uncontrollable modes, we can reduce the size of the *A, B, Q* and *R* matrices that are input to the Riccati equation. The reduction in matrix size significantly lightens the computational load of the Riccati solver. Computation time can be further minimized by choosing a non-adaptive LQR-SM controller which solves the Riccati equation off-line. The excellent performance of the non-adaptive controller is aided by our tight voltage restrictions. If voltage undershoot and overshoot were allowed greater range, the results might have been different.

After optimizing each of the controllers through identical genetic algorithm routines, we found no degradation in the minimum stored energy required by each controller. All four of the controllers (twentieth-order model, seventeenth-order model, eleventh-order model, and non-adaptive twentieth-order model) achieved roughly the same stored energy value.

Examination of the ROAs illustrated that ROA size is highly variable. Although the minimum energy configuration controllers had similar capacitor values for C_{b1}, C_{b2}, and C_{b3}, their ROAs in Zone #1, Zone #2, and Zone #3 were dissimilar to one another, respectively. These differences in ROAs illustrate that although all controllers perform similarly according to the minimum energy fitness function Equation (5.3), they do not perform similarly with regard to ROA size. When the controllers were constrained to have identical penalties for like state variables, the adaptive controllers produced nearly identical ROAs; however, the non-adaptive controller ROAs were different from the adaptive controller ROAs. Sometimes the non-adaptive controller ROAs were larger, other times smaller. If there are specific ROA requirements, those requirements need to be built into the genetic algorithm fitness function.

THIS PAGE INTENTIONALLY LEFT BLANK

VII. CONCLUSIONS

In this dissertation, we set out to find a suitable controller to regulate the voltage of a hypothetical naval MVDC shipboard electrical distribution system. That distribution system would include a zonal architecture, CPL, and HESS.

Background work demonstrated how constant power loads destabilize electrical systems. CPL introduce negative, non-linear impedance. Linearization of CPL impedance has shown that care must be taken in the selection of circuit components in order to avoid unstable eigenvalues. Non-linear analysis demonstrated that even systems that are small-signal stable may not be globally stable. Rather, they exist within ROAs. ROAs can be very sensitive to circuit component selection and CPL loading.

A review of related work found that many authors have addressed the problem of electrical distribution systems with constant power loads but with a very limited focus. The CPL stabilization methods and controllers in the literature all address the problem as a single-input control problem, often simplifying the distribution system into a simple second-order equivalent circuit.

In this work, we addressed a much more general problem than what has previously been addressed in the literature. Rather than limiting the study to a simple problem that could be reduced to a second-order system or single-input control problem, in this work we investigated a complex DC electrical distribution system. The electrical distribution system studied has multiple inputs. Some of those inputs were voltage sources, while others were current sources. Unlike previous studies, the system under investigation could not reasonably be approximated by a second-order system.

A. SIGNIFICANT CONTRIBUTIONS

1. LQR-SM Controller

By expanding the scope of the study to a complex MVDC distribution system, we were forced to employ a multi-input control scheme. The control scheme developed is an adaptive, multi-rate LQR controller that uses selected cost function matrices. The

proposed solution reduces computational complexity and improved robustness as compared to periodic-discrete multi-rate LQR.

The proposed LQR-SM control scheme described in this work was demonstrated through MATLAB simulation to perform with near equivalence to a traditional periodic-discrete implementation of multi-rate LQR. The LQR-SM method of multi-rate LQR produced nearly identical control inputs and system regulation as compared to the classical periodic-discrete implementation but did so using a small fraction of the computing resources. This was done by rotating through a set of either B or R matrices to solve for sub-cycle feedback gains for each possible combination of inputs rather than constructing large block-cyclic matrices. By reducing the size of the matrices operated on by the Riccati equation solver, we reduced the computational load drastically. This reduction on computation load enabled the control designer to employ an adaptive LQR controller rather than be constrained to off-line computation only.

A further advantage of the LQR-SM control scheme was improved scalability and reliability. In Chapter IV Section D, we considered a traditional multi-rate LQR implemented on a thirteenth-order system. Using a thirteenth-order system with an eight-step block-cycle produced enormous block-cyclic A, B, Q, and R matrices. The MATLAB Riccati solver *dare()* issued several warnings regarding ill-conditioned matrices. The *dare()* solver was pushed past its limit when attempting to solve for a larger, twentieth-order circuit model. In contrast, LQR-SM was easily scaled to a twentieth-order circuit system with 16 steps per cycle without issuing any warnings.

While LQR-SM was investigated for a naval MVDC distribution system, the technique described is not constrained to one specific class of problems. The LQR-SM controller can be applied broadly. Whether the system under study is single input or multi-input, small or complex, fully controllable or not, LQR-SM may be applied. The only restrictions are that the system is stabilizable, Q is positive semi-definite, and R is positive definite.

2. Design Process

The next contribution is an easily understood design process. While there are rules of thumb regarding LQR controller design, they are inadequate for high-order problems. When dealing with large and complex systems, the interaction between state-variables and the LQR Q and R matrix values can become overwhelming.

In our design process, the circuit specifications and controller parameters were determined simultaneously. Starting with a passively stable distribution system, the genetic algorithm iterates through the chosen number of generations to select circuit and controller parameter combinations which optimize a fitness function. The genetic algorithm design process described in this dissertation utilized time-domain simulations of a worst-case power transient. The simulation trial results were measured and checked for compliance with overshoot, undershoot, settling time, and harmonic content specifications. Finally, system stored energy was measured, with the optimal candidate having the lowest energy. Other engineers with different or more complex objectives may include those measurements into their own fitness functions.

Due to the CPL in our system, it was imperative that we begin with a passively stable system. Because the system we studied was not globally stable, we could not choose random values for the genomes of the initial generation of our genetic algorithm. Consequently, random selection or even poor selection of the initial generation of genomes may result in an optimization that never converges to a solution within the ROA.

3. Performance Trade-Offs

Lastly, we explored performance trade-offs. Our system had many state variables that closely tracked with each other because impedances were small. We experimented with reduced-order controllers and with a non-adaptive controller to determine what trade-offs, if any, were experienced by reducing the fidelity of the controller or by considering a non-adaptive approach. For this study, we learned that reduced fidelity controllers that ignored small impedances had reduced computational load without any

penalty in minimum energy storage or transient dynamic performance. Even the non-adaptive controller performed as well as the adaptive controllers.

The major differences between the controllers were in the sizes of their respective ROAs. In the minimum energy configuration, ROAs varied quite widely, even though the zone buck capacitors used in each controller scenario were similar. An experiment where controllers used identical circuit parameters and Q and R matrix values revealed that choice of controller Q and R values plays a significant role in the size of ROAs.

B.   FUTURE WORK

So far, LQR-SM controllers have only been implemented in MATLAB software for an average value model. The robustness and utility of the control method could be further proved through implementation on a test bed. Testing the control scheme on a test bed, first as a switch-mode model and then in hardware, could prove the real-world applicability of this method. Furthermore, application to a physical model would also allow for testing the controller's sensitivity to microcontroller delay time and sensor measurement noise. For this to occur, a HESS controlled current source needs to be developed.

In addition, an investigation into ROA size is warranted. Since a system this complex defies analytical analysis of the ROA, a computational investigation into the factors which most strongly influence ROA size would be instructive. The genetic algorithm could be modified to assess ROA size and an ROA score included in the fitness function.

APPENDIX A. MATLAB CODE

1.   Simulated Annealing

```matlab
%Synthetic Annealing Algorithm
%This algorithm randomly modifies the "best performing" iteration of
system
%variables. Initially the variance of change to variables is large, but
%then gradually reduces to settle on a final value.
clear;

Best_Score = 1e6;  %initial best score. Lower score is better.
To = 100;  %initial system variance
T(1) = To; %System temp
k = 1;%number of iterations
iter = 1;
run Circuit_Data25;

C = [Cbus Cd1 Cd2 Cz1 Cz2 Cz3 Cd3 Cb1 5*Cb2 Cb3];
Q = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]; %Q diagonals
R = [1 1 1e3 1 1]; %Generator in use penalty, input unavailable
penalty, Is1 penalty, Is2 penalty

C_best = C; Q_best = Q; R_best = R;
L = [Lg1 Lg2 Lg3 Lg4 Lz1 Lz2 Lz3 Lb1 Lb2 Lb3];
r = [Rg1 Rg2 Rg3 Rg4 Rz1 Rz2 Rz3 Rd1 Rd2 Rd3];
V = [Vref Vref1 Vref2 Vref3];

minValC = [1e-6 1e-7 1e-7 1e-7 Cz1 Cz2 Cz3 Cripple1 Cripple2 Cripple3];
maxValC = C;
minValQ = Q;
maxValQ = 2^11*Q;
minValR = [1 1 1e3 0.05 0.05];
maxValR = [2^4 2^4 1e3 1 1];
minVal = [minValC minValQ minValR];
maxVal = [maxValC maxValQ maxValR];

J(1) = fitness(V, r, L, [C_best Q_best R_best]);
J_best = J(1);

while T > 0
k = k+1;
%Cooling Profile
if k > 2800
T(k) = T(k-1) - 40/400;
elseif k > 1600
T(k) = T(k-1) - 60/1200;
else
T(k) = T(k-1);
end

%Available variables for modification are bus capacitance, zone input
```

```
%filter capacitances, buck capacitances, R-matrix diagonal values,
% and Q-matrix diagonal values.
neighbor = mutateSA([C_best Q_best R_best], minVal, maxVal, T(k), To);

J_neighbor = fitness(V, r, L, neighbor);

if J_neighbor < J_best
J_best = J_neighbor;
C_best = neighbor(1:10);
Q_best = neighbor(11:30);
R_best = neighbor(31:35);
end
J(k) = J_best;
[k T(k) J_best]
end


function [ y ] = mutateSA( seed, minVal, maxVal, T, To )
%MUTATE returns a scalar value based on a random normal distribution
with mean of 'seed', variance of 0.1 and with cut-off minimum and
maximum values.

%This helps to choose only 20% of genes for mutation
chooser = 5*rand(1, length(seed));

for k = 1:length(seed)
if chooser(k) > 1
y(k) = seed(k);
else
y(k) = min(max(minVal(k),  seed(k)*(1 + 0.5*T/To*randn(1)) ),
maxVal(k));
end
end

%enforce that small generators and large generators have same penalties
y(13) = y(11);
y(14) = y(12);
end
```

2.   Genetic Algorithm

3.   Super-Routine

```
%%%Genetic Algorithm - GA with very tight restrictions on C Q R%%%

%This script inputs a circuit parameters file, including all of the RLC
%data for a Four-Generator, Three-Zone MVDC shipboard distribution
system
%with a common MVDC bus.
```

```matlab
%Since the system will be controlled via an adaptively linearizing LQR
%controller, the Q and R matrices will be inputs as well.

%Start with a system that is passively stable, then selectively reduce
%capacitor sizes, increase variable error penalties, and reduce input
%penalties to produce a system with acceptable performance and minimum
%stored energy.

clear;
%clc;

%Import Circuit Data for the passively stable system
run Circuit_Data25;

C = [2*Cbus Cd1 Cd2 Cd3 Cz1 Cz2 Cz3 Cb1 5*Cb2 Cb3];

L = [Lg1 Lg2 Lg3 Lg4 Lz1 Lz2 Lz3 Lb1 Lb2 Lb3];
r = [Rg1 Rg2 Rg3 Rg4 Rz1 Rz2 Rz3 Rd1 Rd2 Rd3];
V = [Vref Vref1 Vref2 Vref3];

ReL = 1; %Input penalty for large generators.
ReS = 1; %Input penalty for small generators.
Pen = 1e3; %Penalty value for unusable inputs.
Rs1 = 1;%Is1 penalty
Rs2 = 1;%Is2 penalty

%Variable Error Penalty Matrix
Q = ones(1,20);

%Input Penalty Matrix
R = [ReL ReS Pen Rs1 Rs2]; %Input penalty for large generator, small
generator, unavailable input, Is1, Is2

%Penalty values
os_penalty = 50;
fft_penalty = 50;

%Define the seed/best
C_best = C;
Q_best = Q;
R_best = R;
J_best = 101;

%Create Generation #1
pop = 16; %population
for i = 1:pop
if i == 1
G(i, :) = [C Q R];
else
minValC = [1e-6 1e-7 1e-7 1e-7 Cz1 Cz2 Cz3 Cripple1 Cripple2 Cripple3];
maxValC = C;
minValQ = Q;
```

```matlab
maxValQ = 2^11*Q;
minValR = [1 1 1e3 0.05 0.05];
maxValR = [2^4 2^4 1e3 1 1];
minVal = [minValC minValQ minValR];
maxVal = [maxValC maxValQ maxValR];
G(i, :) = mutate0([C Q R], minVal, maxVal);
end
end

Generations = 200;
for k = 1:Generations
%tic
%Test fitness of each member of the generation
for i  = 1:pop
Jgen(i) = fitness(V, r, L, G(i, :) );
end
J(k) = min(Jgen)

%Determine which population members are breeders. Here we use top half
m = 1;
for i = 1:pop
if Jgen(i)< median(Jgen)
Breeder(m, :) = G(i, :);
Jbreed(m) = Jgen(i);
m = m+1;
end
end

%Use parents to create children. Parents survive to the next
%generation
for i = 1:length(Jbreed)/2
mixer1 = round(rand(1, length([C Q R])));
mixer2 = 1-mixer1;
parent1 = Breeder(2*i-1, :);
parent2 = Breeder(2*i, :);
child1 = parent1.*mixer1 + parent2.*mixer2;
child2 = parent1.*mixer2 + parent2.*mixer1;
if  Jbreed(2*i-1) == min(Jgen)
%do nothing - this is the elite parent
C_best = parent1(1:10);
Q_best = parent1(11:30);
R_best = parent1(31:35);
else
%mutate parent1
parent1 = mutate(parent1, minVal, maxVal);
end
if Jbreed(2*i) == min(Jgen)
%do nothing - this is the elite parent
C_best = parent2(1:10);
Q_best = parent2(11:30);
R_best = parent2(31:35);
else
%mutate parent2
parent2 = mutate(parent2, minVal, maxVal);
end
```

```matlab
child1 = mutate(child1, minVal, maxVal);
child2 = mutate(child2, minVal, maxVal);

G(2*i-1, :) = parent1;
G(2*i, :) = parent2;
G(pop/2 + 2*i-1, :) = child1;
G(pop/2 + 2*i, :) = child2;
end
%toc
end
```

4.   Mutate Function

```matlab
function [ y ] = mutate0( seed, minVal, maxVal )
%MUTATE returns a scalar value based on a random normal distribution
with mean of 'seed', variance of 0.1 and with cut-off minimum and
maximum values.
for k = 1:length(seed)
y(k) = minVal(k) + (maxVal(k) - minVal(k))*rand(1); %min(max(minVal(k),
seed(k)*(1 + 0.5*randn(1)) ), maxVal(k));
end
%enforce that small generators and large generators have same penalties
y(13) = y(11);
y(14) = y(12);

end


function [ y ] = mutate( seed, minVal, maxVal )
%MUTATE returns a scalar value based on a random normal distribution
with mean of 'seed', variance of 0.1 and with cut-off minimum and
maximum values.

%This helps to choose only 20% of genes for mutation
chooser = 5*rand(1, length(seed));

for k = 1:length(seed)
if chooser(k) > 1
y(k) = seed(k);
else
y(k) = min(max(minVal(k),  seed(k)*(1 + 0.5*randn(1)) ), maxVal(k));
end
end

%enforce that small generators and large generators have same penalties
y(13) = y(11);
y(14) = y(12);
end
```

5.   Fitness Function

```matlab
function [ J_trial ] = fitness( V, r, L, G )
%Run a trial on each population member, then return teh fitness score
%Detailed explanation goes here
%Run a Trial
run Circuit_Data25

os_penalty = 50;
fft_penalty = 50;


C = G(1:10);
Q = G(11:30);
R = G(31:35);


H = trial20(V, r, L, C, Q, R);
Vbus = H(1, :);
Vz1 = H(2, :);
Vz2 = H(3, :);
Vz3 = H(4, :);
Vb1 = H(5, :);
Vb2 = H(6, :);
Vb3 = H(7, :);
E1 = H(8, :);
E2 = H(9, :);
E3 = H(10, :);
E4 = H(11, :);
Is1 = H(12, :);
Is2 = H(13, :);

%%Score the trial

%Overshoot score
J_overshoot = 0;

overshoot_Vbus = max( (max(Vbus)-Vref), abs(min(Vbus) - Vref) )/Vref;
%( max(Vbus) - min(Vbus) )/Vref;
overshoot_Vb1 = max( (max(Vb1)-Vref1), abs(min(Vb1) - Vref1) )/Vref1;
%( max(Vb1) - min(Vb1) )/Vref1;
overshoot_Vb2 = max( (max(Vb2)-Vref2), abs(min(Vb2) - Vref2) )/Vref2;
%( max(Vb2) - min(Vb2) )/Vref2;
overshoot_Vb3 = max( (max(Vb3)-Vref3), abs(min(Vb3) - Vref3) )/Vref3;
%( max(Vb3) - min(Vb3) )/Vref3;

if overshoot_Vbus > 0.10
J_overshoot = J_overshoot + os_penalty;
end
if overshoot_Vb1 > 0.10
J_overshoot = J_overshoot + os_penalty;
end
if overshoot_Vb2 > 0.10
J_overshoot = J_overshoot + os_penalty;
end
if overshoot_Vb3 > 0.10
```

```matlab
J_overshoot = J_overshoot + os_penalty;
end

%Harmonic content score
J_harmonics = 0;
if sldetect(Vbus) > -60
J_harmonics = J_harmonics + fft_penalty;
end
if sldetect(Vb1) > -60
J_harmonics = J_harmonics + fft_penalty;
end
if sldetect(Vb2) > -60
J_harmonics = J_harmonics + fft_penalty;
end
if sldetect(Vb3) > -60
J_harmonics = J_harmonics + fft_penalty;
end

%Capacitance Score
J_capacitance = sum((C - [0 0 0 0 Cz1 Cz2 Cz3 Cripple1 Cripple2
Cripple3]).*[1 1 1 1 0 0 0 (Vref1/Vref) (Vref2/Vref) (Vref3/Vref)]);

%Stored Energy
E_caps = (1/2)*C*[Vref Vref Vref Vref Vref Vref Vref Vref1 Vref2
Vref3].^2';

E_Is1 = zeros(size(Is1));
E_Is2 = zeros(size(Is2));
dt = 0.100/(length(Is1) - 1);

for k = 1:length(Is1)
E_Is1(k) = sum( Vb1(1:k).*Is1(1:k) )*dt;
E_Is2(k) = sum( Vb2(1:k).*Is2(1:k) )*dt;
end
E_Is1_max = max(E_Is1);
E_Is2_max = max(E_Is2);

J_Energy = (E_caps + E_Is1_max + E_Is2_max)/1e6;

%Trial Failure Score
J_tf = 0;
if length(Vbus) < 4000
J_tf = 100;
end

%J_summary = [J_overshoot J_harmonics J_tf J_Energy]

J_trial = J_overshoot + J_harmonics + J_tf + J_Energy; %J_capacitance;


end
```

6.   Trial Function

```matlab
function [YY] = trial20(Vi, ri, Li, Ci, Qi, Ri)

Vref = Vi(1); Vref1 = Vi(2); Vref2 = Vi(3); Vref3 = Vi(4);

Rg1 = ri(1); Rg2 = ri(2); Rg3 = ri(3); Rg4 = ri(4);
Rz1 = ri(5); Rz2 = ri(6); Rz3 = ri(7);
Rd1 = ri(8); Rd2 = ri(9); Rd3 = ri(10);

Lg1 = Li(1); Lg2 = Li(2); Lg3 = Li(3); Lg4 = Li(4);
Lz1 = Li(5); Lz2 = Li(6); Lz3 = Li(7);
Lb1 = Li(8); Lb2 = Li(9); Lb3 = Li(10);

Cbus = Ci(1); Cd1 = Ci(2); Cd2 = Ci(3); Cd3 = Ci(4);
Cz1 = Ci(5); Cz2 = Ci(6); Cz3 = Ci(7);
Cb1 = Ci(8); Cb2 = Ci(9); Cb3 = Ci(10);

Tesd = 1/16e3;
Tgen = 1/4e3;

%State-Variable Initial values

%Circuit_Data25
P1 = [15 5 28]*1e6; %Initial Power for zones 1,2,3
P2 = [20 30 46]*1e6; %Final Power

P = P1;
R = [Vref1 Vref2 Vref3].^2./P;

%Generator Currents
Ig1km1 = 0.40*sum(P)/Vref;%Generator current
Ig2km1 = 0.10*sum(P)/Vref;%Generator current
Ig3km1 = 0.40*sum(P)/Vref;%Generator current
Ig4km1 = 0.10*sum(P)/Vref;%Generator current
Itotalkm1 = Ig1km1+Ig2km1+Ig3km1+Ig4km1;

%Zone Down-stepped values
Ib1km1 = P(1)/Vref1;
Vb1km1 = Vref1;
Vb1km2 = Vb1km1;

Ib2km1 = P(2)/Vref2;
Vb2km1 = Vref2;
Vb2km2 = Vb2km1;

Ib3km1 = P(3)/Vref3;
Vb3km1 = Vref3;
Vb3km2 = Vb3km1;

%Cable Currents
Iz1km1 = (Vref1/Vref)*Ib1km1;%Load current zone 1
Iz2km1 = (Vref2/Vref)*Ib2km1;%Load current zone 2
```

```matlab
Iz3km1 = (Vref3/Vref)*Ib3km1;%Load current zone 3

%Bus Voltage
Vbuskm2 = Vref;%MVDC bus voltage
Vbuskm1 = Vbuskm2;

%Point of Load Voltages
Vz1km2 = Vref;%Voltage at input to zone#1 CPL
Vz1km1 = Vz1km2;


Vd1km2 = Vref;%Damper capacitor voltage
Vd1km1 = Vd1km2;


Vz2km2 = Vref;%Voltage at input to zone#2 CPL
Vz2km1 = Vz2km2;


Vd2km2 = Vref;%Damper capacitor voltage
Vd2km1 = Vd2km2;


Vz3km2 = Vref;%Voltage at input to zone#3 CPL
Vz3km1 = Vz3km2;


Vd3km2 = Vref;%Damper capacitor voltage
Vd3km1 = Vd3km2;


%Input Device Values
E1km2 = Vref + Rg1*Ig1km1;%Generator voltage
E1km1 = E1km2;
E2km2 = Vref + Rg2*Ig2km1;%Generator voltage
E2km1 = E2km2;
E3km2 = Vref + Rg3*Ig3km1;%Generator voltage
E3km1 = E3km2;
E4km2 = Vref + Rg4*Ig4km1;%Generator voltage
E4km1 = E4km2;


Is1km1 = 0;%Zone 1
Is2km1 = 0;%Zone 2
Is3km1 = 0;


dt = 5e-9;
t = 0:dt:2*1e7*dt;
%decimate = length(t)/400;


tau = 0.05;
Pz1estkm1 = P(1);
Pz2estkm1 = P(2);
Pz3estkm1 = P(3);
Pestkm1 = sum(P);


Rb1 = 1; Rb2 = 1; Rb3 = 1;
d1 = Vref1/Vref; d2 = Vref2/Vref; d3 = Vref3/Vref;


%Pen = 1e3;%Penalty for unusable inputs;
```

```matlab
index = 1;
decimate = 5e3;
token = 0;


for k = 1:length(t)


Pz1 = Vb1km1*( Ib1km1 - Cb1*(Vb1km1 - Vb1km2)/dt + Is1km1);
Pz2 = Vb2km1*( Ib2km1 - Cb2*(Vb2km1 - Vb2km2)/dt + Is2km1);
Pz3 = Vb3km1*( Ib3km1 - Cb3*(Vb3km1 - Vb3km2)/dt + Is3km1);


if mod(t(k), Tesd) == 0
%Implement LQR
Pz1estk = tau*Pz1 + (1-tau)*Pz1estkm1;%Zone power estimates
Pz2estk = tau*Pz2 + (1-tau)*Pz2estkm1;
Pz3estk = 10*tau*Pz3 + (1-10*tau)*Pz3estkm1;
Pestk = Pz1estk + Pz2estk + Pz3estk;


I0 = Pestk/Vref; %Equilibrium current value
Io1 = Pz1estk/Vref;
Io2 = Pz2estk/Vref;
Io3 = Pz3estk/Vref;
Iob1 = Pz1estk/Vref1;
Iob2 = Pz2estk/Vref2;
Iob3 = Pz3estk/Vref3;


%CPL non-linear resistance values
Rnl1 = -Vb1km1^2/Pz1estk; %-Vz1km1^2/Pz1estk;
Rnl2 = -Vb2km1^2/Pz2estk; %-Vz2km1^2/Pz2estk;
Rnl3 = -Vb3km1^2/Pz3estk; %-Vz3km1^2/Pz3estk;



X1 = Ig1km1 - .40*I0; %Change of variables
X2 = Ig2km1 - .10*I0;
X3 = Ig3km1 - .40*I0;
X4 = Ig4km1 - .10*I0;
X5 = Vbuskm1 - Vref;
X6 = Iz1km1 - Io1;
X7 = Iz2km1 - Io2;
X8 = Iz3km1 - Io3;
X9 = Vz1km1 - Vref - Rz1*Io1;
X10 = Vz2km1 - Vref - Rz2*Io2;
X11 = Vz3km1 - Vref - Rz3*Io3;
X12 = Vd1km1 - Vref - Rz1*Io1;
X13 = Vd2km1 - Vref - Rz2*Io2;
X14 = Vd3km1 - Vref - Rz3*Io3;
X15 = Ib1km1 - Iob1;
X16 = Ib2km1 - Iob2;
X17 = Ib3km1 - Iob3;
X18 = Vb1km1 - Vref1;
X19 = Vb2km1 - Vref2;
X20 = Vb3km1 - Vref3;
```

```matlab
X = [X1; X2; X3; X4; X5; X6; X7; X8; X9; X10; X11; X12; X13; X14; X15;
X16; X17; X18; X19; X20];

%ESDs switching but generators are not
if mod(t(k), Tgen) ~= 0
mode = 0;
U = SysControl20(X,d1,d2,d3, Rnl1,Rnl2,Rnl3, Li, ri, Ci, Qi, Ri, mode);
Is1k = U(18);
Is2k = U(19);
Is3k = U(20);
end

% ESDs and One generator are switching together
if mod(t(k), Tgen) == 0
if token == 0
mode = 1;
U = SysControl20(X,d1,d2,d3, Rnl1,Rnl2,Rnl3, Li, ri, Ci, Qi, Ri, mode);
E1k = U(1) + Vref + Rg1*0.40*I0;
E2k = E2km1; %U(2) + Vref + Rg2*0.10*I0; %E2km1;
E3k = E3km1; %U(3) + Vref + Rg3*0.40*I0; %E3km1;
E4k = E4km1; %U(4) + Vref + Rg4*0.10*I0; %E4km1;
Is1k = U(18);
Is2k = U(19);
Is3k = U(20);
token = 1;
elseif token == 1
mode = 2;
U = SysControl20(X,d1,d2,d3, Rnl1,Rnl2,Rnl3, Li, ri, Ci, Qi, Ri, mode);
E1k = E1km1; %U(1) + Vref + rg1*0.40*I0(k);
E2k = U(2) + Vref + Rg2*0.10*I0;
E3k = E3km1; %U(3) + Vref + rg3*0.40*I0(k);
E4k = E4km1; %U(4) + Vref + rg4*0.10*I0(k);
Is1k = U(18);
Is2k = U(19);
Is3k = U(20);
token = 2;
elseif token == 2
mode = 3;
U = SysControl20(X,d1,d2,d3, Rnl1,Rnl2,Rnl3, Li, ri, Ci, Qi, Ri, mode);
E1k = E1km1; %U(1) + Vref + rg1*0.40*I0(k);
E2k = E2km1; %U(2) + Vref + rg2*0.10*I0(k);
E3k = U(3) + Vref + Rg3*0.40*I0;
E4k = E4km1; %U(4) + Vref + rg4*0.10*I0(k);
Is1k = U(18);
Is2k = U(19);
Is3k = U(20);
token = 3;
elseif token == 3
mode = 4;
U = SysControl20(X,d1,d2,d3, Rnl1,Rnl2,Rnl3, Li, ri, Ci, Qi, Ri, mode);
E1k = E1km1; %U(1) + Vref + rg1*0.40*I0(k);
E2k = E2km1; %U(2) + Vref + rg2*0.10*I0(k);
E3k = E3km1; %U(3) + Vref + rg3*0.40*I0(k);
E4k = U(4) + Vref + Rg4*0.10*I0;
Is1k = U(18);
```

```matlab
Is2k = U(19);
Is3k = U(20);
token = 0;
end
end
end


Is3k = 0;

%State-Space equations
Id1k = (Vz1km1 - Vd1km1)/Rd1;
Id2k = (Vz2km1 - Vd2km1)/Rd2;
Id3k = (Vz3km1 - Vd3km1)/Rd3;



dIg1 = (1/Lg1)*(E1k - Rg1*Ig1km1 - Vbuskm1);
dIg2 = (1/Lg2)*(E2k - Rg2*Ig2km1 - Vbuskm1);
dIg3 = (1/Lg3)*(E3k - Rg3*Ig3km1 - Vbuskm1);
dIg4 = (1/Lg4)*(E4k - Rg4*Ig4km1 - Vbuskm1);
dVbus = (1/Cbus)*(Ig1km1 + Ig2km1 + Ig3km1 + Ig4km1 - Iz1km1 - Iz2km1 -
Iz3km1);
dIz1 = (1/Lz1)*(Vbuskm1 - Rz1*Iz1km1 - Vz1km1);
dIz2 = (1/Lz2)*(Vbuskm1 - Rz2*Iz2km1 - Vz2km1);
dIz3 = (1/Lz3)*(Vbuskm1 - Rz3*Iz3km1 - Vz3km1);
dVz1 = (1/Cz1)*( Iz1km1 - Id1k - d1*Ib1km1);
dVz2 = (1/Cz2)*( Iz2km1 - Id2k - d2*Ib2km1);
dVz3 = (1/Cz3)*( Iz3km1 - Id3k - d3*Ib3km1);
dVd1 = (1/Cd1)*Id1k;
dVd2 = (1/Cd2)*Id2k;
dVd3 = (1/Cd3)*Id3k;
dIb1 = (1/Lb1)*(d1*Vz1km1 - Vb1km1);
dIb2 = (1/Lb2)*(d2*Vz2km1 - Vb2km1);
dIb3 = (1/Lb3)*(d3*Vz3km1 - Vb3km1);
dVb1 = (1/Cb1)*(Ib1km1 - P(1)/Vb1km1 + Is1k );
dVb2 = (1/Cb2)*(Ib2km1 - P(2)/Vb2km1 + Is2k );
dVb3 = (1/Cb3)*(Ib3km1 - P(3)/Vb3km1 + Is3k );

%Variable updates
Ig1k = Ig1km1 + dIg1*dt;
if Ig1k < 0
Ig1k = 0;
end
Ig2k = Ig2km1 + dIg2*dt;
if Ig2k < 0
Ig2k = 0;
end
Ig3k = Ig3km1 + dIg3*dt;
if Ig3k < 0
Ig3k = 0;
end
Ig4k = Ig4km1 + dIg4*dt;
if Ig4k < 0
Ig4k = 0;
```

114

```matlab
end
Vbusk = Vbuskm1 + dVbus*dt;
Iz1k = Iz1km1 + dIz1*dt;
Iz2k = Iz2km1 + dIz2*dt;
Iz3k = Iz3km1 + dIz3*dt;
Vz1k = Vz1km1 + dVz1*dt;
Vz2k = Vz2km1 + dVz2*dt;
Vz3k = Vz3km1 + dVz3*dt;
Vd1k = Vd1km1 + dVd1*dt;
Vd2k = Vd2km1 + dVd2*dt;
Vd3k = Vd3km1 + dVd3*dt;
Ib1k = Ib1km1 + dIb1*dt;
if Ib1k < 0
Ib1k = 0;
end
Ib2k = Ib2km1 + dIb2*dt;
if Ib2k < 0
Ib2k = 0;
end
Ib3k = Ib3km1 + dIb3*dt;
if Ib3k < 0
Ib3k = 0;
end
Vb1k = Vb1km1 + dVb1*dt;
Vb2k = Vb2km1 + dVb2*dt;
Vb3k = Vb3km1 + dVb3*dt;

%Break out of the trial if it appears care() will fail
if (Vb1k/Vref1) < 0.1
break
elseif (Vb2k/Vref2) < 0.1
break
elseif (Vb3k/Vref3) < 0.1
break
end


Vz1km2 = Vz1km1;
Vz2km2 = Vz2km1;
Vz3km2 = Vz3km1;
Vb1km2 = Vb1km1;
Vb2km2 = Vb2km1;
Vb3km2 = Vb3km1;

Ig1km1 = Ig1k;
Ig2km1 = Ig2k;
Ig3km1 = Ig3k;
Ig4km1 = Ig4k;
Iz1km1 = Iz1k;
Iz2km1 = Iz2k;
Iz3km1 = Iz3k;
Vbuskm1 = Vbusk;
Vz1km1 = Vz1k;
Vz2km1 = Vz2k;
Vz3km1 = Vz3k;
```

```matlab
Vd1km1 = Vd1k;
Vd2km1 = Vd2k;
Vd3km1 = Vd3k;
Ib1km1 = Ib1k;
Ib2km1 = Ib2k;
Ib3km1 = Ib3k;
Vb1km1 = Vb1k;
Vb2km1 = Vb2k;
Vb3km1 = Vb3k;


Pz1estkm1 = Pz1estk;
Pz2estkm1 = Pz2estk;
Pz3estkm1 = Pz3estk;


E1km1 = E1k;
E2km1 = E2k;
E3km1 = E3k;
E4km1 = E4k;
Is1km1 = Is1k;
Is2km1 = Is2k;
Is3km1 = Is3k;


%decimation routine
if mod(k, decimate) == 0
Ig1(index) = Ig1k;%Generator Currents
Ig2(index) = Ig2k;
Ig3(index) = Ig3k;
Ig4(index) = Ig4k;
Vbus(index) = Vbusk;%Main bus voltage
Iz1(index) = Iz1k;%Zone cable currents
Iz2(index) = Iz2k;
Iz3(index) = Iz3k;
Vz1(index) = Vz1k;%Voltages at input to buck converters
Vz2(index) = Vz2k;
Vz3(index) = Vz3k;
Vd1(index) = Vd1k;%Damper capacitor voltages
Vd2(index) = Vd2k;
Vd3(index) = Vd3k;
Id1(index) = Id1k;%Damper capacitor currents
Id2(index) = Id2k;
Id3(index) = Id3k;
Ib1(index) = Ib1k;%Buck chopper currents
Ib2(index) = Ib2k;
Ib3(index) = Ib3k;
Vb1(index) = Vb1k;%Buc chopper voltages
Vb2(index) = Vb2k;
Vb3(index) = Vb3k;
E1(index) = E1k;%Generator VOltages
E2(index) = E2k;
E3(index) = E3k;
E4(index) = E4k;
Is1(index) = Is1k;%Injected Currents
Is2(index) = Is2k;
Is3(index) = Is3k;
Pest(index) = Pestk;%Total Estimated load power
```

```matlab
Pz1est(index) = Pz1estkm1;%Zone estimated load power
Pz2est(index) = Pz2estkm1;
Pz3est(index) = Pz3estkm1;
Iob1_(index) = Iob1;%Desired buck chopper current
Iob2_(index) = Iob2;
Iob3_(index) = Iob3;
time(index) = t(k);
%PP(index) = P;
%Icpl(index) = P/Vz1k;
%Vest(index) = Vestk;
index = index + 1;
end


%Change load power
if ( t(k) > 0.001)&&( t(k) < 0.050 )
P = P2;
R = [Vref1 Vref2 Vref3].^2./P;
elseif t(k) > 0.050
P = P1;
R = [Vref1 Vref2 Vref3].^2./P;
end


end


YY = [Vbus; Vz1; Vz2; Vz3; Vb1; Vb2; Vb3; E1; E2; E3; E4; Is1; Is2];
```

7.   Controller Function

```matlab
function [U] = SysControl20(X,d1,d2,d3, Rnl1,Rnl2,Rnl3, Li, ri, Ci, Qi,
Ri, mode);
%This function to be used in conjuction with CPL case studies
Rg1 = ri(1); Rg2 = ri(2); Rg3 = ri(3); Rg4 = ri(4);
Rz1 = ri(5); Rz2 = ri(6); Rz3 = ri(7);
Rd1 = ri(8); Rd2 = ri(9); Rd3 = ri(10);

Lg1 = Li(1); Lg2 = Li(2); Lg3 = Li(3); Lg4 = Li(4);
Lz1 = Li(5); Lz2 = Li(6); Lz3 = Li(7);
Lb1 = Li(8); Lb2 = Li(9); Lb3 = Li(10);

Cbus = Ci(1); Cd1 = Ci(2); Cd2 = Ci(3); Cd3 = Ci(4);
Cz1 = Ci(5); Cz2 = Ci(6); Cz3 = Ci(7);
Cb1 = Ci(8); Cb2 = Ci(9); Cb3 = Ci(10);

Pen = Ri(3);
if mode == 0
r_1 = Pen;
r_2 = Pen;
r_3 = Pen;
r_4 = Pen;

elseif mode == 1
r_1 = Ri(1);
```

117

```matlab
r_2 = Pen;
r_3 = Pen;
r_4 = Pen;

elseif mode == 2
r_1 = Pen;
r_2 = Ri(2);
r_3 = Pen;
r_4 = Pen;

elseif mode == 3
r_1 = Pen;
r_2 = Pen;
r_3 = Ri(1);
r_4 = Pen;

elseif mode == 4
r_1 = Pen;
r_2 = Pen;
r_3 = Pen;
r_4 = Ri(2);

end

B11 = 1; B22 = 1; B33 = 1; B44 = 1;

A = [-Rg1/Lg1 0 0 0 -1/Lg1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 -Rg2/Lg2 0 0 -1/Lg2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 -Rg3/Lg3 0 -1/Lg3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 -Rg4/Lg4 -1/Lg4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
1/Cbus 1/Cbus 1/Cbus 1/Cbus 0 -1/Cbus -1/Cbus -1/Cbus 0 0 0 0 0 0 0 0 0
0 0 0;...
0 0 0 0 1/Lz1 -Rz1/Lz1 0 0 -1/Lz1 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 1/Lz2 0 -Rz2/Lz2 0 0 -1/Lz2 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 1/Lz3 0 0 -Rz3/Lz3 0 0 -1/Lz3 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 1/Cd1 0 0 -1/Cd1/Rd1 0 0 1/Cd1/Rd1 0 0 -d1/Cd1 0 0 0 0 0;...
0 0 0 0 0 0 1/Cd2 0 0 -1/Cd2/Rd2 0 0 1/Cd2/Rd2 0 0 -d2/Cd2 0 0 0 0 0;...
0 0 0 0 0 0 0 1/Cd3 0 0 -1/Cd3/Rd3 0 0 1/Cd3/Rd3 0 0 -d3/Cd3 0 0 0 0;...
0 0 0 0 0 0 0 0 1/Cd1/Rd1 0 0 -1/Cd1/Rd1 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 1/Cd2/Rd2 0 0 -1/Cd2/Rd2 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 1/Cd3/Rd3 0 0 -1/Cd3/Rd3 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 d1/Lb1 0 0 0 0 0 0 0 0 -1/Lb1 0 0;...
0 0 0 0 0 0 0 0 0 0 d2/Lb2 0 0 0 0 0 0 0 0 -1/Lb2 0;...
0 0 0 0 0 0 0 0 0 0 0 d3/Lb3 0 0 0 0 0 0 0 0 -1/Lb3;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1/Cb1 0 0 1/(Cb1*Rnl1) 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1/Cb2 0 0 1/(Cb2*Rnl2) 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1/Cb3 0 0 1/(Cb3*Rnl3)];

B = [B11/Lg1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 B22/Lg2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 B33/Lg3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 B44/Lg4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
```

118

```matlab
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1/Cb1 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1/Cb2 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];


Q = [Qi(1) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%Ig1
0 Qi(2) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%Ig2
0 0 Qi(3) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%Ig3
0 0 0 Qi(4) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%Ig4
0 0 0 0 Qi(5) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%Vbus
0 0 0 0 0 Qi(6) 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%Iz1
0 0 0 0 0 0 Qi(7) 0 0 0 0 0 0 0 0 0 0 0 0 0;...%Iz2
0 0 0 0 0 0 0 Qi(8) 0 0 0 0 0 0 0 0 0 0 0 0;...%Iz3
0 0 0 0 0 0 0 0 Qi(9) 0 0 0 0 0 0 0 0 0 0 0;...%Vz1
0 0 0 0 0 0 0 0 0 Qi(10) 0 0 0 0 0 0 0 0 0 0;...%Vz2
0 0 0 0 0 0 0 0 0 0 Qi(11) 0 0 0 0 0 0 0 0 0;...%Vz3
0 0 0 0 0 0 0 0 0 0 0 Qi(12) 0 0 0 0 0 0 0 0;...%Vd1
0 0 0 0 0 0 0 0 0 0 0 0 Qi(13) 0 0 0 0 0 0 0;...%Vd2
0 0 0 0 0 0 0 0 0 0 0 0 0 Qi(14) 0 0 0 0 0 0;...%Vd3
0 0 0 0 0 0 0 0 0 0 0 0 0 0 Qi(15) 0 0 0 0 0;...%Ib1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Qi(16) 0 0 0 0;...%Ib2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Qi(17) 0 0 0;...%Ib3
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Qi(18) 0 0;...%Ib3
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Qi(19) 0;...%Ib3
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Qi(20)];...%Ib3
%Variable error penalty matrix

R_ = [r_1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%E1
0 r_2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%E2
0 0 r_3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%E3
0 0 0 r_4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...%E4
0 0 0 0 Pen 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 Pen 0 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 Pen 0 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 Pen 0 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 Pen 0 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 Pen 0 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 Pen 0 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 Pen 0 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 Pen 0 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 Pen 0 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 Pen 0 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Pen 0 0 0 0;...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Pen 0 0 0;...
```

```matlab
            0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Ri(4) 0 0;...%Is1 0.1 0.25
            0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Ri(5) 0;...%Is2 0.3 0.50
            0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 Pen]; %Input penalty matrix

try
[XX, LL, GG] = care(A, B, Q, R_); %Riccati Solver
catch
GG = zeros(size(A));
end


U = -GG*X; %Input vector
end
```

8.   Harmonic Side-Lobe Detector

```matlab
%%%Sidelobe detection routine%%%
function [SLL] = sldetect(Q)
% %Get the maginitude FFT of the desired signal
mag = abs(fft(Q));
%normalize the FFT
norm = mag./(max(mag));
db = 20*log10(norm);
if length(db) > 100
SLL = max(db(30:100));
else
SLL = 1;
end
end
```

APPENDIX B. SEED CANDIDATE CALCULATIONS

```
%Circuit Data for case Study 25
%Circuit Parameters
Vref = 12e3; %Main Bus voltage
Vref1 = 1e3; %Zone 1 voltage
Vref2 = 6e3; %Zone 2 voltage
Vref3 = 10e3; %Zone 3 voltage
Rg1 = 4*62.5e-3; %Ohm
Lg1 = 4*500e-6; %Henry
Rg2 = 4*75e-3;%Ohm
Lg2 = 4*450e-6;%Henry
Rg3 = 4.1*62.5e-3; %Ohm
Lg3 = 3.9*500e-6; %Henry
Rg4 = 4.2*75e-3;%Ohm
Lg4 = 3.8*450e-6;%Henry
Cbus = 500e-5; %6.5e-3;%Farad


P2 = [20 30 46]*1e6;
P1 = [15 5 28]*1e6;


fswitch_gen = 4.0e3;
Tgen = 1/fswitch_gen;
fswitch_ESD = 4*fswitch_gen;
Tesd = 1/fswitch_ESD;
Fsw_buck = 1e3;


%Cable Variables
%Based on the Cupelli, de Carro, Monti 2015 paper
%For 800mm cross section cable, rated for 871 Amps, 22.1uOhm/m,
470nH/m,
%4100pF/m
%8000A/871 => 10 parallel cables
rho = 22.1e-6; %per-meter reistance
ind = 470e-9; %per meter inductance
cap = 4100e-12; %per meter capacitance

%Zone#1-20MW Ship's Service
N1 = ceil(P2(1)/Vref/871);
Len = 300; %length of cable in meters
Rz1 = Len*rho/N1; %Ohm
Lz1 = Len*ind/N1; %Henry
Cz1 = Len*cap*N1; %Faraday

d1 = Vref1/Vref;
Req1 = Vref1^2/P1(1);

Lb1 = Req1/(2*Fsw_buck)*(1-d1);
Cripple1 = (1-d1)/Fsw_buck^2/(8*Lb1*.05);

%Zone#2-30MW Pulsed Loads
N2 = ceil(P2(2)/Vref/871);
```

121

```matlab
Len = 300; %length of cable in meters
Rz2 = Len*rho/N2; %Ohm
Lz2 = Len*ind/N2; %Henry
Cz2 = Len*cap*N2; %Faraday


d2 = Vref2/Vref;
Req2 = Vref2^2/P1(2);


Lb2 = Req2/(2*Fsw_buck)*(1-d2);
Cripple2 = (1-d2)/Fsw_buck^2/(8*Lb2*.05);

%Zone#3-80MW Populsion
N3 = ceil(P2(3)/Vref/871);
Len = 300; %length of cable in meters
Rz3 = Len*rho/N3; %Ohm
Lz3 = Len*ind/N3; %Henry
Cz3 = Len*cap*N3; %Faraday


d3 = Vref3/Vref;
Req3 = Vref3^2/P1(3);


Lb3 = Req3/(2*Fsw_buck)*(1-d3);
Cripple3 = (1-d3)/Fsw_buck^2/(8*Lb3*.05);


taubus = max([Lg1/Rg1 Lg2/Rg2 Lg3/Rg3 Lg4/Rg4]);
Reqbus = Vref^2/sum(P2);
Cbus = taubus/Reqbus; %0.0039; %Farad
Cd1 = P2(1)/sum(P2)*Cbus;
Cd2 = P2(2)/sum(P2)*Cbus;
Cd3 = P2(3)/sum(P2)*Cbus;


Rg = (1/Rg1 + 1/Rg2 + 1/Rg3 + 1/Rg4)^-1;
Z1 = (1/d1^2)*Vref1^2/P2(1) + Rz1;
Z2 = (1/d2^2)*Vref2^2/P2(2) + Rz2;
Z3 = (1/d3^2)*Vref3^2/P2(3) + Rz3;


Zin1 = (d1^2)*( Rz1 + (Rg^-1 + Z2^-1 + Z3^-1)^-1 );
Zin2 = (d2^2)*( Rz2 + (Rg^-1 + Z1^-1 + Z3^-1)^-1 );
Zin3 = (d3^2)*( Rz3 + (Rg^-1 + Z1^-1 + Z2^-1)^-1 );


C1 = Lb1/Req1/Zin1;
C2 = Lb2/Req2/Zin2;
C3 = Lb3/Req3/Zin3;


Cb1 = max(Cripple1, C1);
Cb2 = max(Cripple2, C2);
Cb3 = max(Cripple3, 4*C3);


Cx = [Cbus Cd1 Cd2 Cd3 Cb1 Cb2 Cb3];


z = 10;
Rd1 = z; Rd2 = z; Rd3 = z;
C = [Cbus Cd1 Cd2 Cd3 Cb1 Cb2 Cb3];
```

LIST OF REFERENCES

[1]   J. Kuseian, "Naval power systems technology development roadmap," Naval Sea Systems Command, Washington, DC, PMS 320, Apr. 2013.

[2]   N. Doerry, "Medium voltage dc power for the future fleet," presented at the Captain Ralph R. and Florence Peachman Lecture, Ford Presidential Library, University of Michigan Naval Architecture and Marine Engineering, Ann Arbor, Michigan, April 15, 2015.

[3]   A. Riccobono and E. Santi, "Comprehensive review of stability criteria for dc power distribution systems*," IEEE Transactions on Industrial Applications*, vol. 50, no. 5, pp. 3525–3535, September/October, 2014.

[4]   J. Hou, J. Sun, and H. Hofmann, "Interaction analysis and integrated control of hybrid energy storage and generator control system for electric ship propulsion," in *American Control Conference*, Chicago, IL, 2015, pp. 4988–4993.

[5]   A. T. Elsayed and O. A. Muhammed, "A comparative study on the optimal combination of hybrid energy storage system for ship power systems," in *IEEE Electric Ship Technologies Symposium*, Alexandria, VA, 2015, pp140-144.

[6]   A. Mohamed, V. Salehi, and O. Muhammed, "Real-time energy management algorithm for mitigation of pulse loads in hybrid microgrids," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1911–1922, December 2012.

[7]   M. Farhadi and O. Muhammed, "Adaptive energy management in redundant hybrid DC microgrid for pulse load mitigation," *IEEE Transactions on Smart Grid* , vol. 6, no. 1, pp. 54–62, January 2015.

[8]   N. Doerry, J. Amy, and C. Krolick, "History and the status of electric ship propulsion, integrated power systems, and future trends in the U.S. Navy," *Proceedings of the IEEE*, vol. 103, no. 12, pp. 2243–2251, December 2015.

[9]   "The electric warship with an integrated power system and a zonal electric distribution system," class notes for EC3150, Dept. of Electrical and Computer Engineering, Naval Postgraduate School, Monterey, CA, Autumn, 2015.

[10]   N. Mohan, T. M. Undeland, and W. P. Robbins, *Power Electronics: Converters, Applications and Design*, 3rd edition, Hoboken, NJ: John Wiley & Sons, 2003.

[11]   C. G. Hodge, J. O. Flower, and A. Macalinden, "DC power system stability," in *IEEE Electric Ship Technologies Symposium*, 2009, Baltimore, MD, pp. 433–439.

[12]   V. Arcidiacono, A. Monti, and G. Sulligoi, "Generation control system for improving design and stability of medium-voltage dc power systems on ships," *IET Electrical Systems in Transportation*, vol. 2, no. 3, pp. 158 – 167, October, 2012.

[13]   P. Magne, D. Marx, B. Nahid-Mobarakeh, and S. Pierfederici, "Large-signal stabilization of a DC-link supplying a constant power load using a virtual capacitor: Impact on the domain of attraction," *IEEE Transactions on Industry Applications*, vol. 48, no. 3, pp. 878–887, May/June, 2012.

[14]   A. Kwasinski and C. Onwuchekwa, "Dynamic behavior and stabilisatization of DC microgrids with instantaneous constant power loads," *IEEE Transactions on Power Electronics,* vol. 26, no. 3, pp. 822–834, March, 2011.

[15]   M. Zadeh, R. Gavagsaz-Ghoachani, S. Pierfederici, B. Nahid-Mobarakeh, and M. Molinas, "Stability analysis and dynamic performance evaluation of a power electronics-based DC distribution system with active stabilizer," *IEEE Journal of Emerging and Selected Topics in Power Engineering*, vol. 4, no. 1, pp. 93–102, March, 2016.

[16]   M. Zadeh, R. Gavagsaz-Ghoachani, J-P. Martin, S. Pierfederici, B. Nahid-Mobarakeh, and M. Molinas, "Discrete-time modelling, stability analysis, and active stabilization of DC distribution systems with constant power loads," in *IEEE Applied Power Electronics Conference and Exposition,* Charlotte, NC, 2015*, pp. 323–329.

[17]   S. Grillo; V. Musolino, G. Sulligoi, and E. Tironi, "Stability enhancement in DC distribution systems with constant power controlled converters," in *IEEE 15th International Conference on Harmonics and Quality of Power,* Hong Kong, China, 2012, pp. 848–854.

[18]   M. S. Carmeli; D. Forlani, S. Grillo, R. Pinetti, E. Ragaini, and E. Tironi , "A stabilization method for dc networks with constant-power loads," in *IEEE International Energy Conference and Exhibition,* Florence, Italy, 2012, pp. 617–622.

[19]   V. Arcidiacono, A. Monti, and G. Sulligoi, "an innovative generation control system for improving design and stability of shipboard medium-voltage DC integrated power system," in *IEEE Electric Ship Technologies Symposium,* Baltimore, MD, 2009, pp. 152–156.

[20]   J. G. Ciezki and R. W. Ashton, "The application of feedback linearization techniques to the stabilization of DC-to-DC converters with constant power loads," *International Symposium on Circuits and Systems*, Monterey, CA, vol. 3, 1998, pp. 526–529.

[21]   G. Sulligoi, D. Bosich, L. Zhu, M. Cupelli, and A. Monti, "linearizing control of shipboard multi-machine mvdc power systems feeding constant power loads," *IEEE Energy Conversion Congress and Exposition,* Raleigh, NC, 2012, pp. 691–697.

[22]   G. Sulligoi, D. Bosich, G. Giadrossi, L. Zhu, M. Cupelli, and A. Monti, "Multiconverter medium voltage DC power systems on ships: Constant-power loads instability solution using linearization via state-feedback control," *IEEE Transactions on Smart Grid*, vol. 5, no. 5, pp. 2543–2552, September , 2014.

[23]   G. Sulligoi; D. Bosich, and G. Giadrossi, "linearizing voltage control of mvdc power systems feeding constant power loads: Stability analysis under saturation," *IEEE Power and Energy Society General Meeting,* 2013, pp. 1–5.

[24]   G. Sulligoi, D. Bosich, V. Arcidiacono, and G. Giadrossi, "considerations on the design of voltage control for multi-machine mvdc power systems on large ships," *IEEE Electric Ship Technologies Symposium,* Arlington, VA, 2013, pp. 314–319.

[25]   D. Bosich, G. Giadrossi, and G. Sulligoi, "Voltage control solutions to face the CPL instability in MVDC Shipboard power systems," *Associazione Italiana di Elettrotecnica Annual Conference (AEIT)*, Trieste, Italy, 2014, pp. 1–6.

[26]   L. Zhu, J. Liu, M. Cupelli, and A. Monti, "decentralized linear quadratic gaussian control of multi-generator mvdc shipboard power system with constant power loads," *IEEE Electric Ship Technologies Symposium,* Arlington, VA, 2013, pp. 308–313.

[27]   M. Cupelli, M. de Paz Carro, and A. Monti, "hardware in the loop implementation of a disturbance based control in MVDC grids," *IEEE Power & Energy Society General Meeting*, 2015, pp. 1–5.

[28]   M. Cupelli, M. Mirz, and A. Monti, "Application of backstepping to MVDC ship power systems with constant power loads," *International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles, Aachen, Germany*, 2015, pp. 1– 6.

[29]   M. Cupelli, M. Mirz, and A. Monti, "A comparison of backstepping and LQG control for stabilizing MVDC microgrids with constant power loads," *IEEE PowerTech*, Eindhoven, Netherlands, 2015, pp. 1– 6.

[30]   I. Kondratiev and R. Dougal, "Synergetic control strategies for shipboard DC power distribution systems," *American Control Conference*, New York, NY, 2007, pp. 4744–4749.

[31]   I. Kondratiev and R. Dougal, "Current distribution control design for paralleled DC/DC converters using synergetic control theory," *IEEE Power Electronics Specialists Conference*, Orlando, FL, 2007, pp. 851–857.

[32]   M. Cupelli, M. Moghimi, A. Riccobono, and A. Monti, "A comparison between synergetic control and feedback linearization for stabilizing MVDC microgrids with constant power load," *5th IEEE PES Innovative Smart Grid Technologies Europe*, Istanbul, Turkey, 2014, pp. 1–6.

[33]   N. Doerry and J. V. Amy Jr., "Design considerations for a reference MVDC power system," presented at SNAME Maritime Convention 2016, Bellevue, WA, Nov 1–5, 2016.

[34]   K. Mizokami, 2016 Oct 19, The Zumwalt destroyer is here, now what about the railgun? *Popular Mechanics*. [Online] Available at http://www.popularmechanics.com/military/ navy-ships/a23440/zumwalt-destroyer-railgun/

[35]   All Hands Magazine, USS Zumwalt. [Online]. Available at http://www.navy.mil/ah_online/zumwalt/

[36]   D. E. Kirk, *Optimal Control Theory: An Introduction,* Mineola, NY: Dover Publications Inc., 1998.

[37]   K. Natarajan and R. Balasubrumanian, "On the design of LQR multirate controllers," *American Control Conference*, Seattle, WA, 1986, pp.1974–1979.

[38]   A. J. Mills and R. W. Ashton, "Adaptive, sparse, and multi-rate LQR control of an MVDC shipboard power system with constant power loads," *IEEE International Conference on Industrial Technology*, Toronto, Ontario, Canada, 2017, pp. 498–503.

[39]   A. J. Mills and R. W. Ashton, "Multi-rate LQR control of a multi-machine MVDC shipboard electric distribution system with constant power loads," *IEEE Midwest Symposium on Circuits and Systems*, Medford, MA, 2017, publication pending.

[40]   E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*, Hoboken, NJ: John Wiley & Sons, 1989.

[41]   R. L. Haupt and S. E. Haupt*, Practical Genetic Algorithm*, Hoboken, NJ: John Wiley and Sons, 2004.

[42]   A. J. Mills and R. W. Ashton, "Genetic Algorithm Design of an Adaptive, Multirate LQR Controller for a Multi-Machine MVDC Shipboard Electric Distribution System with Constant Power Loads," *IEEE International Transportation Electrification Conference*, Chicago, IL, 2017, publication pending.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center
	Ft. Belvoir, Virginia

2.	Dudley Knox Library
	Naval Postgraduate School
	Monterey, California